AFIT/GLM/LAL/96S-7

EVALUATION OF *THE GOAL SYSTEM*™ VERSION 2.2 SOLUTION METHOD FOR INTERACTIVE CONSTRAINT SCHEDULING SITUATIONS

THESIS

Michael D. Stephens, Captain, USAF

AFIT/GLM/LAL/96S-7

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

AFIT/GLM/LAL/96S-7

EVALUATION OF *THE GOAL SYSTEM*™ VERSION 2.2 SOLUTION METHOD FOR INTERACTIVE CONSTRAINT SCHEDULING SITUATIONS

Presented to the Faculty of the School of Logistics and Acquisition Management

of the Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Logistics Management

Michael D. Stephens, B.S.
Captain, USAF

September 1996

Approved for public release; distribution unlimited

Acknowledgments

I would like to thank my advisor, Lieutenant Colonel Jacob Simons, for performing a vital role in this research effort. His familiarity with the previous research, his guidance throughout the thesis process, and his genuine concern for my success made this research possible.

Although she was not able to contribute directly to the research, I credit my wife, Yong Hui Stephens, with maintaining an environment supportive of my efforts. Aside from my heartfelt thanks, I can not compensate her for the many sacrifices she made for the good of this thesis.

Michael D. Stephens

Table of Contents

Page

Page

List of Figures

List of Tables

Abstract

THE GOAL SYSTEM™ version 2.2 is the latest in a lineage that includes Optimized Production Technology (OPT) and *DISASTER*™. Earlier work with *DISASTER*™ revealed potential shortcomings with sequential schedule-building algorithms when multiple interactive constraints exist. Since *THE GOAL SYSTEM*™ version 2.2 has a capacity for simultaneous schedule-building, this study evaluated differences between the two algorithms. Using benchmark scheduling problems developed during the earlier evaluation of *DISASTER*™, a set of *THE GOAL SYSTEM*™ solutions was created and compared quantitatively to both *DISASTER*™ solutions and solutions which optimally minimize maximum tardiness. A broad set of performance measurement criteria were also used to obtain a more comprehensive evaluation of the solutions. Performance of *THE GOAL SYSTEM*™ was quite good with respect to maximum tardiness. Performance with respect to average flow-time, percentage of tardy jobs, and total days late for a set of job orders was markedly poorer than the *DISASTER*™ solutions. The results were unexpected, since the simultaneous scheduling algorithm is less restricted in its options for schedule creation. The author concluded that the simultaneous feature of *THE GOAL SYSTEM*™ was better suited for conflict resolution during an iterative process than as a stand-alone scheduling algorithm.

EVALUATION OF *THE GOAL SYSTEM*™ VERSION 2.2 SOLUTION METHOD FOR INTERACTIVE CONSTRAINT SCHEDULING SITUATIONS

I. Problem Description

General Issue

Eliyahu Goldratt's ideas concerning production management have been in use successfully since the late 1970's. Among other titles, his ideas are known as the Theory of Constraints (TOC). The U.S. Air Force has widely adopted TOC ideas to improve aircraft modification processes at its Air Logistics Centers (Carlson and Lettiere, 1993:1).

To implement TOC in scheduling a manufacturing system, it is first necessary to understand the concept of a constraint. Assuming that the goal of the manufacturing operation is to make money, constraints are defined as elements that prevent the system from making more money (Umble and Srikanth, 1990:81). Umble and Srikanth note that each system contains one or more constraints (1990:81). Consider the outcome if this were not true. A system without constraints would be capable of generating an infinite amount of money. Since there can't be infinite demand for any product, the system will always be faced with a market constraint. Umble and Srikranth categorize other constraints as material, capacity, logistical, managerial, and behavioral types (1990:91).

In building a production schedule in accordance with TOC, the idea is to concentrate on exploiting the productivity of capacity constraint resources, while subordinating other resources to the constraints. Constraint resources can be exploited by eliminating unnecessary idle time and minimizing wasted production by the resource.

For example, parts should not be routed to a constraint resource if they are in danger of eventually failing quality checks for processes which occurred upstream from the constraint resource. Faulty parts should be identified and removed prior to routing to the constraint resource. Subordination of non-constraint resources demands that these resources produce only enough materials to keep the constraint resources busy, despite their possible capacity to produce more. While this may seem to be willful under-utilization of those resources, the alternative is an ever-increasing amount of work-in-process waiting to be processed by the constraint resource. The correct level of production for non-constraint resources is one that matches the level of production for downstream constraint resources.

Evolutionary software packages have been created to build production schedules using the TOC philosophy. This philosophy is operationalized in a method of scheduling known as the Drum-Buffer-Rope method. The first of the series was known as Optimized Production Technology (OPT), released around 1979. Early successes of the OPT software were reported by large corporations like General Electric and General Motors (Simons and Simpson, 1995:1). In 1990, the Avraham Y. Goldratt Institute released the *DISASTER*™ software package. As of 1994, the TOC Center gained the rights to *DISASTER*™, and promptly renamed it *THE GOAL SYSTEM*™ (TGS).

In a significant departure from previous policy, the *DISASTER*™ algorithm was not kept proprietary (Simons and others, 1995:10). United States Air Force Captains Stewart W. James and Bruno A. Mediate published the logic of *DISASTER*™ in thesis AFIT/GSM/LAS/93S-9. They also created a bank of 108 benchmark scheduling

problems to be used in evaluating the performance of a variety of scheduling algorithms. The focus of their research was the *DISASTER*™ software's technique of dealing with multiple constraints sequentially. Since one constraint is scheduled first, with subsequent constraint schedules subject to the timing restrictions of the first, the final schedule stands to be sub-optimized. (Carlson and Lettiere, 1993:4).

There are various means for measuring the quality of a schedule. Many feel that due-date performance is the outstanding indicator of a schedule's usefulness (Carlson and Lettiere, 1993:3). Some performance measures which fall under the due-date heading are: total days late, maximum tardy days, and percentage of job orders tardy. The distinction between tardiness and lateness with respect to due-date performance is that tardiness can never be a value of less than zero, while lateness may have a positive or negative value. The objectives of a particular production system determine the relative importance of the different performance measures. Not all performance measures can be simultaneously optimized (James and Mediate, 1993:13). Carlson and Lettiere concluded that the heuristic algorithm used by *DISASTER*™ performed strongly in terms of minimizing maximum tardiness on the majority of James and Mediate's 108 test problems (1993:58). This was verified by comparing the maximum tardiness of solutions created by *DISASTER*™ to optimal solutions provided by their simultaneous scheduling algorithm. Their comparison did not extend to other measures of performance, however.

Specific Problem

TGS software, including its three modules; NETGEN, CALENDAR, and SCHEDULE, has been updated since James and Mediate and Carlson and Lettiere evaluated the 1993 version, known at that time as *DISASTER*™. The latest TGS release is version 2.2. It is significant to note that in version 2.2, the sequential approach to scheduling multiple constraints is still available, but the software now includes a simultaneous scheduling function as well. While sequential scheduling was a frequently criticized characteristic of *DISASTER*™ (Carlson and Lettiere, 1993:4), the extent of improvement to schedule quality using the new algorithm has not been established. Neither has the algorithm of the new package been published.

Hypothesis

TGS version 2.2 scheduling software, using a simultaneous scheduling algorithm, could be expected to produce a higher quality schedule for production scenarios involving interactive multiple constraints than the earlier *DISASTER*™ software, using a sequential algorithm.

Investigative Questions

1. What in particular has changed in the algorithm between *DISASTER*™ and TGS version 2.2?
2. In terms of maximum tardiness performance, how, if at all, has the quality of schedules produced by TGS software changed with respect to solutions created by *DISASTER*™ and in relation to optimal solutions?

3. How do *DISASTER*™ and *THE GOAL SYSTEM*™ version 2.2 compare in terms of alternate performance criteria such as total days late, percentage of tardy job orders, and average flow time?

Key Variables

Scheduling Software

DISASTER™, *THE GOAL SYSTEM*™, and the optimal solution algorithm of Carlson and Lettiere.

Performance Measures

Maximum Tardy Days, Percentage of Tardy Job Orders, Total Days Late, and Average Flow Time

Scope

This research effort is limited to the software's performance with respect to solving the benchmark scheduling problems. These problems were designed to represent three specific types of plant operations with two constraint resources at several levels of constraint loading. The problems do not address real-world factors such as setup times and dynamic job order arrivals. Work-in-process is not measured directly, but is roughly indicated by flow times through the plant.

Methodology

The research will begin with a literature review to establish familiarity with methods and results of prior research in the area of scheduling and evaluation of scheduling techniques. The operation of the software will then be observed to identify relevant differences between software versions. The 108 benchmark scheduling problems will then be scheduled with *THE GOAL SYSTEM*™ software and the results

compared to those of *DISASTER*™ and the optimal solution algorithm of Carlson and Lettiere.

Summary

This chapter provided an overview of the nature of and motivation for this research. Chapter II provides a more detailed background of previous research on the topic. The literature review also brings forth concepts fundamental to understanding scheduling and the evaluation of scheduling techniques. Chapter III presents the methodology used in this research effort. Chapter IV documents the results and analysis of the comparison between scheduling algorithms. Finally, Chapter V summarizes the research effort and recommends potential avenues for future research.

II. Literature Review

Introduction

This chapter reviews significant background concerning the theory and operation of *DISASTER*™ scheduling software. A detailed investigation of the fundamentals of *DISASTER*™ is called for because of its role as predecessor to *THE GOAL SYSTEM*™ (TGS). An understanding of *DISASTER*™ is necessary so that differences between it and TGS can be recognized. Tools like benchmark problems and performance measures can be used to evaluate the software, and are reviewed here as well. A glossary of terms appears in Appendix A.

Theory of Constraints

Production scheduling can be a difficult task, complicated by many variable factors which must be balanced in order to identify the "best" of possible schedules. Often, managers can become fixated on particular aspects of the production chain, to the detriment of the system as a whole. The revolutionary principles popularized by Israeli physicist Eliyahu Goldratt have provided managers a whole new way to think about production. It is important to understand that production is not an end unto itself, but rather a necessary link in a chain of efforts conducted to meet some organizational goal. Typically this goal is earning profit through sales of manufactured goods. This new way of thinking is known, among other names, as the Theory of Constraints (TOC).

An analogy taken from Umble and Srikanth (1990:54-60) illustrates the basis of the TOC philosophy. A simple production process can be thought of in terms of a

column of soldiers on a forced march. To duplicate a process in which production functions are dependent on the functions that precede them, imagine a column of soldiers marching in single file. Before a soldier can cross a particular piece of ground, the ground must first have been covered by all preceding soldiers. Ground that has been walked by the first soldier but not the last is analogous to work-in-process, or inventory. Ground remaining to be walked by all of the soldiers is the raw material, while ground that has been walked by the entire column is the finished product.

Over time, the column of soldiers tends to spread out because of statistical variations in marching speed. In the long run, fluctuations in the first soldier's speed tend to balance each other out because his walking speed varies between faster than average and slower than average. As the marching speed of ensuing soldiers fluctuates, gaps form between the soldiers whenever the speed of the following soldier is less than that of the soldier in front. However, when the following soldier's marching speed would exceed that of the soldier in front, gains are limited by the pace of the soldier in front. Posterior soldiers can walk infinitely slower than the soldiers in front, but they are limited their ability to walk faster. So the result of statistical variations in marching speed is a tendency for the column of soldiers to spread out over time. In production terms, "The maximum deviation of a preceding operation will become the starting point of a subsequent operation." (Goldratt and Cox, 1986:133). Even if all soldiers are able to maintain a comparable average speed, statistical variations in individual speeds over time will cause this spreading-out phenomenon.

Now consider the case where all soldiers do not have equal ability with relation to marching speed. You can see that the slowest soldier in the column determines the rate that ground is covered by the entire troop. If preceding soldiers exceed his pace, the result is that the column spreads out. But the last man doesn't get to the finish line any faster. If the arrival time of the last soldier is unsatisfactory, then the pace of the slowest soldier has not met the demand placed on him. Since he can't meet the demand placed on him, he is a constraint to the system. These processes and terms apply to manufacturing as well, and are the basis for the Theory of Constraints.

Intuition may incorrectly dictate that the most efficient way to operate a manufacturing shop is to maximize the efficiency of each of the components of the process. It can be demonstrated that local improvements can potentially have no impact on throughput of the system. Worse yet, they might even have a negative impact. Think back to the soldiers analogy. Each individual soldier marching at his top speed doesn't necessarily get the last man to the finish any faster if he is a constraint or is impeded by a constraint. "Local optima do not add up to the optimum of the total." according to Goldratt (1990:51).

The objectives of the Theory of Constraints method for managing a system are summarized by five focusing steps.

The Five Focusing Steps of TOC (Goldratt, 1990:58-63)

1. *Identify the system's constraint.* Discover which resources are the weak links in the production chain. Which resources have greater demands on them than they have capacity?

2. *Exploit the system's constraint.* Maximize the usefulness of the constraint. Do not allow it to sit idle. Prioritize the products that have access to the constrained resource, and reroute those of lowest priority.

3. *Subordinate everything else to the constraint.* Processes upstream from the constraint must limit their production to levels required to keep the constraint in operation, but no higher. Excessive production upstream to maximize that unit's efficiency will only result in a backlog at the constraint resource, and an increase in work-in-process inventory.

4. *Elevate the system's constraint.* Increase the capacity of the constraint. For example, consider buying another machine that can perform the constrained function. This should only be considered after the first three steps have been accomplished.

5. *Don't let inertia set in* (Iterate back to step 1). When steps 1 through 4 have successfully relieved a constraint, remain vigilant for the appearance of others.

As applied to the management of production systems, TOC ideals are manifested in the Drum-Buffer-Rope (DBR) approach (Goldratt and Fox, 1986:98). The term "drum" comes from the requirement to set a pace. The drum beat is determined by the constraint's capacity and timing. All resources are driven by it. Buffers, allowable work-in-process, protect against fluctuations in the flow of materials through the system, and are thought of in terms of time rather than materials. Buffers are placed before critical operations, not at the source of disruptions (Goldratt and Fox, 1986:112). If product flow toward the constraint was interrupted, and there was no buffer, system throughput would be interrupted. By definition, a constraint does not have the capacity to catch up, so the buffer is needed. Non-constraint resources do not require buffers because they can utilize their excess capacity to catch up. A constraint buffer is required to protect the constraint from fluctuations in upstream production, while a shipping buffer protects shipping dates from disruptions in non-constraints following the final constraint

operation. Finally, a rope is a method for restraining non-constraint resources from over-production. The rope controls the rate at which raw materials are input to the system. By limiting the release of raw materials to the amount scheduled by the drum in the next buffer time frame, the rope ensures that the non-constraint resources remain subordinate to the drum (Schragenheim and Ronen, 1990:19).

DBR Scheduling Software

When the TOC philosophy is practiced, schedules are only produced for critical resources. As reported by Simons and Simpson (1995:1), DBR scheduling software has existed in various forms since the late 1970's. Eliyahu Goldratt's first software scheduling package was produced in 1979 with the cooperation of Creative Output Inc. This Optimized Production Technology (OPT) software, as it was called, was credited along with its underlying TOC philosophy for remarkable reductions in lead time and inventory in such corporations as General Electric and General Motors. Although Goldratt's renown increased during the mid-eighties with the publication of his 1984 book, *The Goal*, his software and philosophy were largely ignored by academics. The proprietary nature of OPT restricted access to Goldratt's algorithm and his customers– obstacles that academics simply could not surmount (Simons and Simpson, 1995:2). OPT was a mainframe-based computer program that cost over $500,000 for some companies to implement (Severs, 91:4).

In 1990, Goldratt's educationally-focused Avraham Y. Goldratt Institute released another software package, *DISASTER*™. The new program was microcomputer-based,

rather than mainframe-based. Although this time Goldratt was not so secretive about the algorithm of the program, it continued to receive very little interest from academics. A splinter organization known as The TOC Center broke away from the Avraham Y. Goldratt Institute and obtained the rights to educational products and the *DISASTER*™ software (Simons and Simpson, 1995:3). Their revised version of the scheduling software is known as *THE GOAL SYSTEM*™ (TGS). A characteristic of *DISASTER*™ software was that when confronted with multiple interactive constraints, i.e. more than one constraint required to process the same product, it dealt with them by scheduling each constraint sequentially.

Two thesis teams at the Air Force Institute of Technology investigated the *DISASTER*™ software (James and Mediate, 1993; Carlson and Lettiere, 1993). Their work included the creation of a set of benchmark production problems and an optimal scheduling algorithm. These tools allowed them to evaluate the quality of schedules produced by the heuristics of *DISASTER*™ software.

The quality of a production schedule can be measured in many ways. Major categories would be cost and performance (Graves, 1981:648). While cost-related measures such as work-in-process inventory and equipment utilization are interesting and somewhat important, due-date performance measures are of primary interest to practitioners seeking to fill delivery promises on time (Conway and others, 1967:229). James and Mediate chose to evaluate *DISASTER*™ schedules based on maximum tardy days for a set of job orders, and total days late for a set of job orders. A limitation of the original *DISASTER*™ software's sequential scheduling method was identified in that the

choice of the order in which multiple constraints were scheduled sometimes affected the quality of the schedules produced.

In 1995, the TOC Center released *THE GOAL SYSTEM*™ version 2.2. This latest version has the ability to deal with multiple interactive constraints either simultaneously or sequentially. This feature is not highly touted in the SCHEDULE user's guide. There have been no academic research studies performed to evaluate the success of the simultaneous algorithm in comparison with the sequential one. In general, there has been a lack of significant research concerning finite capacity backward scheduling methods (Lalsare and Sen, 1995:71).

Conceptual flow of the DISASTER™ Algorithm

A macro-level description of the *DISASTER*™ algorithm has been reported by Simons and Simpson (1996:13-23). Goldratt devotes Part Three of his 1986 book, *The Haystack Syndrome*, to describing the Drum-Buffer-Rope scheduling methodology which is implemented by *DISASTER*™ and *THE GOAL SYSTEM*™.

DBR scheduling performs the functions of the first three TOC focusing steps, listed above. Constraint identification is accomplished through rough-cut capacity checking. Exploitation of the constraint is accomplished through the drum-building process. Finally, subordination is a final rough-cut capacity check to ensure that non-constraints have sufficient capacity to keep the pace set by the drum. If not, the program loops back and identifies additional constraint(s). Table 1 presents the logical sequence (adapted from Simons and Simpson, 1996:13-15).

Table 1: Conceptual Flow of *DISASTER*™ Logic

Step 1		Compute effective horizon by adding one shipping buffer to the planning horizon
Step 2		Subordinate all resources to the market
	2a	Perform rough-cut capacity check, working backwards from the end of the effective horizon
	2b	Identify resource constraints via First Day Load (FDL) peaks
	2c	If no resource constraints are identified, go to Step 7
Step 3		Build drum schedule for primary constraint resource
	3a	Build ruins
	3b	Perform backward pass to level the ruins
	3c	If batches are scheduled earlier than the present, perform forward pass to achieve feasibility
	3d	Fix drum schedule in time and reconcile constraint batch times with order due-dates
Step 4		Subordinate non-constraint resources to the market and the drum schedule(s)
	4a	Reaccomplish rough-cut capacity check for non-constraint resources (as in Step 2a)
	4b	Identify additional constraints via FDL peaks or Red Lane peaks
	4c	If no additional resource constraints are identified, go to Step 7
Step 5		Build schedule for additional drum
	5a	Build, then level ruins (as in Steps 3a-d), respecting time and batch rods
	5b	Identify drum violations
	5c	If no drum violations exist, return to Step 4
Step 6		Drum Loop
	6a	Rebuild the first fixed constraint schedule, shifting batches later by the amount of the drum violation
	6b	Eliminate all additional constraint schedules
	6c	Go to Step 4
Step 7		Stop--Implement drum schedules

Computation of the effective horizon is an important first step in the process. When developing a schedule up through a given planning horizon, we cannot stop with jobs that fall due within the planning horizon. If a job is due one day after the planning horizon ends, that's not the time to begin work on the job. So adding a shipping buffer to establish an effective horizon gives us a peek at what lies beyond the planning horizon so we can schedule accordingly.

The initial rough-cut capacity check determines whether our resources can meet the demands of the market without declaring any constraints. If so, there is no need to build drums for specific resources. If no constraints are identified up front, we simply treat the market as a constraint. Subordination to the market determines whether non-constraint resources have the capacity to meet the demands of the market. If so, we need not develop dedicated schedules for particular resources.

The need for the creation of a drum will be signaled by the existence of a First-Day Load (FDL) peak. An FDL peak is the need for more capacity on the first day of the schedule than a particular resource has available. If building a drum for a resource is found to be necessary, the first step is identifying the amount of time required of the resource and where it fits on the time axis. The interval of time required for each order can be represented by a block placed on a timeline. Proper placement is such that the end of the processing interval occurs one shipping buffer prior to the job's due-date. Blocks are initially placed according to ideal timing, without concern for resource capacity. For this reason, the blocks may tend to stack up in a rather jumbled fashion. What we have created is a form of Gantt chart which is named by Goldratt (1990:204) as the "ruins." Its resemblance to ruins can be seen in Figure 1.

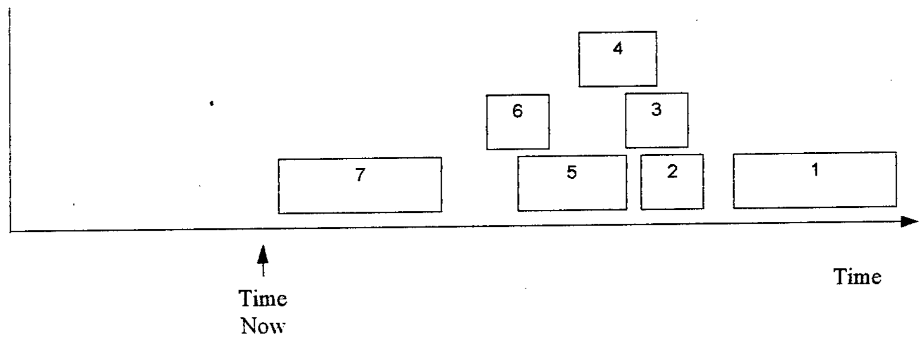

Figure 1: The Ruins

Because the ruins depicted in Figure 1 are three deep, the ideal solution would be to have three units of this resource. If there are less than three units available, the ruins will have to be leveled. In this example, consider that only one unit of this resource is available. Leveling is accomplished with a backward pass across the ruins, placing loads earlier in time. As seen in Figure 2, the blocks are still in due-date sequence, but no more than one unit deep.

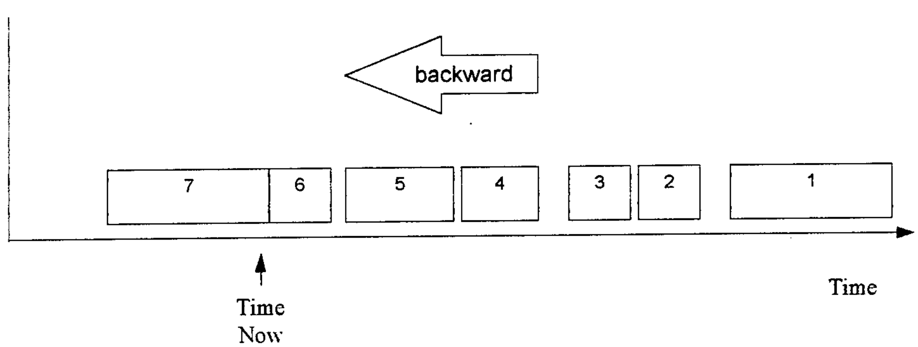

Figure 2: Leveling the Ruins

The backward pass has caused another problem. Block 7 has been moved so early that it must now be accomplished in the past (prior to Time Now). Since this is not possible, a forward pass will be necessary, and can be seen in Figure 3. As block 7 is

pushed forward in time, it contacts other blocks and pushes them forward too. Depending on the structure of the ruins, blocks may be pushed forward so far that they infringe on the shipping buffer or completely miss their due-dates. While late job orders are not desirable, the forward pass is necessary to make our schedule feasible. Following the forward pass, the resulting placement of the blocks is our constraint schedule, or drum.

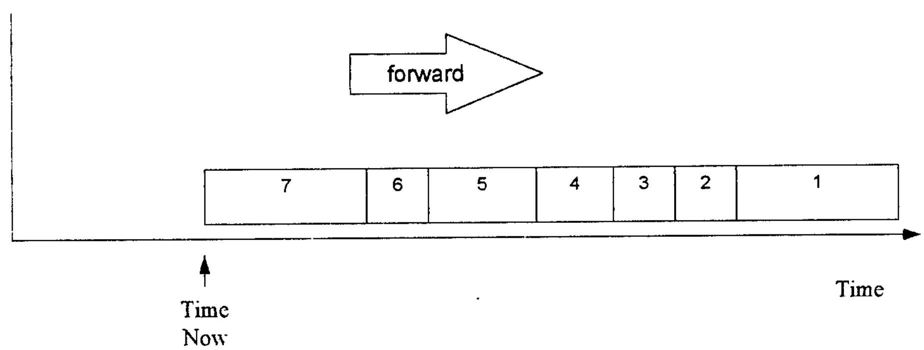

Figure 3: Creating the Drum

An issue that remains to be discussed is the possibility that several of the blocks in the drum may represent succeeding tasks of a single order. This condition in itself does not cause a conflict, and the blocks are free to touch one another as seen in Figure 3. But if processing by non-constraint resources is required between the two constraint processes, the schedule is once again infeasible unless time to perform the non-constraint processing is reserved in the drum. A rod can be attached to a block as required to provide protection forward in time, backward in time, or in both directions.

The duration of rods is set heuristically at one-half the length of the constraint buffer. This provides some measure of protection, but does not provide the expensive

luxury of a complete buffer (Goldratt, 1990: 218). The rod does not interfere with the movement of the blocks during forward and backward passes along the time dimension, but dictates that the adjacent blocks will move as far as necessary to maintain the minimum gap between constraint operations. Figure 4 illustrates the placement of rods under two different situations. Case A shows two different batches, each composed of six parts to be processed. Since the first operation is lengthier than the second, the rod is attached so that the final part of the second operation can not be scheduled earlier than one half of a constraint buffer after the final part in the first operation is complete. By maintaining the relationship between the last two parts of each operation, the others parts are guaranteed even greater protection.

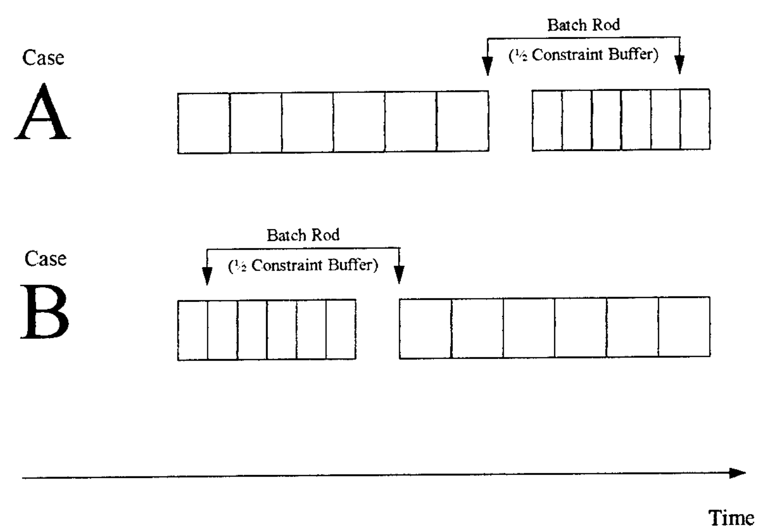

Figure 4: Batch Rods

If the same technique were used in case B, you can see that the second operation would be able to overlap the first operation. So in cases where the first operation is shorter than the second, the rod is attached so that the start time of the first part in the

second operation is separated from the completion time of the first part in the first operation by at least one-half of a constraint buffer. Again, this provides even greater protection for the remaining parts.

Figure 5 shows a variation of the situation shown in case A of Figure 4. Obviously two batches on a single resource can not be scheduled to overlap because the resource is only capable of processing one part at a time. But if the constraint operations are on two different machines, then overlapping is acceptable as long as the one-half buffer of space is maintained. Rods between batches on different drums are called time rods rather than batch rods.

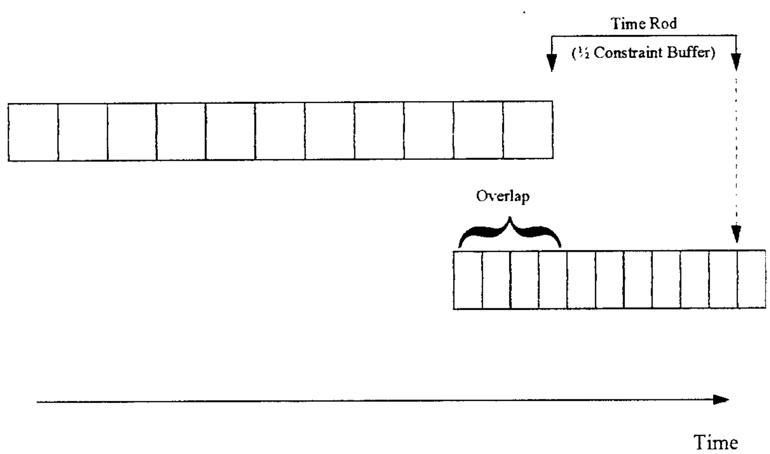

Figure 5: Time Rods

After the drum has been created, *DISASTER*™ makes another subordination run. As before, the presence of FDL peaks among the non-constraint resources indicates that an additional constraint must be declared. This can sometimes be avoided with intervention from the operator, but such techniques are not discussed here.

It is possible that if an additional constraint is declared, earlier drums may not have left sufficient room for the placement of the new drum's batches and rods. This situation is known as a drum violation, and must be resolved by looping back and adjusting the offending drum according to the magnitude of the violation. Again, the scheduler has options beyond the scope of this discussion for resolving drum violations. Whether or not a drum violation occurs, the process always ends with subordination.

Schedule Performance Measures

There are various means for measuring the quality of a schedule. Many feel that due-date performance is the outstanding indicator of a schedule's usefulness (Carlson and Lettiere, 1993:3). In measuring the performance of schedules, simple measurements pertaining to completion-time, flow-time, lateness, and tardiness are often used (Conway and others, 1967:12). Regular measures of performance are those which will increase if even one completion-time increases, a feature common to all of the measures just mentioned. Averages and maximums of these simple measures are frequently used, as are weighted averages, aggregated maximums and averages, and functions of fractiles (Conway and others, 1967:12). The distinction between tardiness and lateness with respect to due-date performance is that tardiness can never be a value of less than zero, while lateness may have a positive or negative value.

Cost-based measures of performance provide important insight to the quality of a schedule as well. Work-in-process (WIP) is a popular measure of schedule performance. There are many ways to calculate WIP, and it has a strong relationship with mean flow

time. Conway and others (1967:20) report that WIP and mean flow-time are directly proportional.

Benchmark Problems

The benchmark problems created by Captains James and Mediate were designed and tested to be representative of diverse scheduling situations. While there are many potential variables to be considered in real-world scheduling, the benchmark problems are differentiated in terms of three main variables, with efforts taken to control other background variables.

The first variable is plant type. The benchmark problems are all based on either V, A, or T plant types. These major categories of manufacturing environments classify plants according to the dominant resource/product interactions they exhibit . (Umble and Srikanth, 1990:210). V-Plants are characterized by divergent operations. They create relatively diverse types of finished goods from a relatively narrow group of raw materials. A-Plants are convergent operations, creating relatively few types of finished goods from diverse raw materials. T-Plants share several traits with both of the other plant types, yet are a distinct category. T-Plants can be defined as an assemble-to-order operation. Several different component parts are assembled to produce multiple types of finished goods. The components making up one finished good are often integral to other types of finished goods as well. The plant layouts used in the benchmark problems can be seen in Appendix B.

The second variable is Resource Criticality Factor (%RCF) expressed as a percentage. This number is a measure of the demand placed on a resource as a percentage of its available capacity (Gargeya, 1992:3). The benchmark problems seek to create three levels of %RCF in the lesser of the two constrained resources: 105 percent, 115 percent, and 125 percent (James and Mediate, 1993:35).

The third variable refers to the %RCF of the greater constrained resource. It was operationalized by James and Mediate as the percentage of difference between the lesser constrained resource and the greater constrained resource. The benchmark problems represent differences in %RCF at three levels: null, 25 percent, and 50 percent.

Creating benchmark problems with each possible combination of the above factors resulted in 27 problem types. Each of these 27 types was replicated four times, using randomly selected due-dates, resulting in 108 benchmark problems.

The benchmark problems contained background variables which were mostly controlled by holding them constant. Each benchmark plant contains 10 dissimilar resources, of which only one of each type is available. Each problem contains two bottleneck resources, with %RCF greater than 100%. Non-constraint resources were targeted for a %RCF of 25%. The time horizon was two weeks in all cases. Each benchmark problem involved scheduling 10 jobs, with each job consisting of a quantity of 100 units. Setup times were held constant at zero and all buffers were calculated to be 8 hours in duration. The location of constraint and non-constraint resources within the plant was held constant for all cases of a particular plant type. The locations were

necessarily different between plant types. Protective capacity was held constant at 5% for all non-constraints.

Summary

This chapter reviewed the essential background for understanding the comparisons intended by this research effort. The major points of Goldratt's Theory of Constraints were presented, along with the relationship of Drum-Buffer-Rope scheduling to TOC. The evolution of DBR software was reviewed and a macro-level discussion of *DISASTER*™ logic was presented. Tools which will be helpful in performing the comparisons between TGS and *DISASTER*™ were also described.

The following chapter reveals the methodology used to collect and analyze relevant data in the remainder of the study.

III. Methodology

Introduction

Inasmuch as the intent of the research was a comparison of *THE GOAL SYSTEM*™ to *DISASTER*™, the researcher remained faithful to the greatest extent possible to the methodology used by James and Mediate in their creation of *DISASTER*™ solutions. There were minor cases where this was not possible or not practical. This chapter presents the methodology used.

Assumptions

1. All buffers are 8 hours long, and sufficient to prevent starvation of constraint resources.

2. All rods are based on one half of buffer length, and are sufficient to permit necessary non-constraint processes between constraint schedules.

3. Blue and Gold are the only resources that will be identified as constraints.

4. Only one resource exists of each type. Resources are labeled Blue, Gold, Black, White, Green, Yellow, Orange, Cyan, Pink, and Red.

5. Resources are not interchangeable and can only process one batch at a time.

Data Collection

By following the procedures outlined in the SCHEDULE user's guide, the typical scheduling sequence would match the one described by Simons and Simpson (1996). The simultaneous algorithm would come into play only if three conditions were met. First, multiple interactive constraints would have to be identified by the SCHEDULE

module. Second, a drum violation would have to occur with an earlier drum. Finally, the scheduler would have to select the *Recreate Old+New Drums* option to fix the violation. These conditions would not all be present for every benchmark problem. Since the objective of this research was to observe the performance of the simultaneous algorithm, the researcher took the necessary steps to schedule each benchmark problem using the simultaneous algorithm. This section describes, in sequence, the steps taken to build schedules using the TGS simultaneous algorithm

Prior to processing the problems with the SCHEDULE module, the researcher used the text-based files created by James and Mediate as inputs to *DISASTER*™'s NETGEN 2.2.5 to create Tasks Structure Net (*.NET) input files for TGS. This was necessary because the *.NET files used by James and Mediate with *DISASTER*™ were incompatible with the TGS SCHEDULE 2.2.8 module. James and Mediate's Calendar (*.CAL) file was not available, so the TGS CALENDAR 2.1.2. module was used to create a file which duplicated James and Mediate's calendar, consisting of a Monday through Friday operation, eight hours per day, with no overtime.

Upon starting the SCHEDULE module, the first screen in the scheduling process was the Parameter Screen. During setup of parameters, the planning horizon used by James and Mediate was increased to begin on 10/03/93 and end on 11/13/93. The reason for using this longer horizon was the nuance reported by James and Mediate concerning the undocumented characteristic of *DISASTER*™ to not fully schedule jobs whose completion dates exceed the effective horizon (1993:80-82). All buffers were set at eight hours. CALENDAR module inputs established the Work Hours per Day parameter, so no

further input was necessary. No overtime was authorized, and Protective Capacity was set at the minimum of five percent. After updating the Parameter file, the researcher selected the Identification Screen. Here, the effect of the lengthened planning horizon could be seen. Initial rough-cut capacity calculations indicated that resources were less heavily loaded than if the calculation had been made using the shorter horizon. Proportional relationships in percentage-of-load differences between resources were still visible via the Resource List, used to display the results of the rough-cut capacity calculation. This screen provided the opportunity to verify key pieces of data about resource loading and, therefore, capacity constraint and control point selection (SCHEDULE User's Guide, 1995:6-1).

The results of the rough-cut capacity calculations were presented on the Identification screen in a selectable resource list. By selecting *Explore,* then *Resource List* from the menu, the author was given authority to scroll through, and highlight any of the resources on the list by pressing the Up/Down Arrow keys on the keyboard. At this point, the author used an undocumented technique to select both the blue resource and the gold resource for drum building. Insight to the availability of this technique was provided by Mr. Rob Newbold, former Director of The TOC Center's Software Support Group. By highlighting the desired resources one at a time, then pressing the Control and Enter keys simultaneously, a check mark was placed beside the resource. After both blue and gold had been checked in this manner, the Escape key enabled the researcher to return to menu-driven options. The researcher selected *Subordinate* from the menu, and selected *Gold+* when prompted to verify his choice for a constraint. Selecting *Gold+*

caused SCHEDULE to build a single ruins for both the gold and blue resources, then simultaneously build drums for each.

The typical procedure would be to select no resources, choose *Subordinate* from the menu upon exiting the Parameter Screen, and select *Market* when prompted to verify choice of a constraint. In the case of the benchmark problems, blue and gold would be identified by the SCHEDULE module as having FDL peaks. The user would then select a single resource from a list of resources found, through subordination, to have FDL peaks. This would lead to the same iterative procedure used in the *DISASTER*™ algorithm. At this point, selecting multiple resources as constraints would not be an option unless the user started over from the Parameter Screen.

After the blue and gold constraints were selected, the procedure continued as described in the previous chapter's conceptual description of the *DISASTER*™ algorithm until a final schedule was produced. Since drums were built for the blue and gold resources in a single iteration, there was never an opportunity for a drum violation to occur.

Performance Measures

Data used in calculating the performance measures originated in every case from the New Order Due Dates (*.SD3) files created for each benchmark problem by the SCHEDULE modules of both *DISASTER*™ and *THE GOAL SYSTEM*™. Specifically, they were derived from fields 4, 5, and 7 of the *.SD3 file. Field 4 provided order quantities, field 5 contained a late/on-time flag, and field 7 was labeled by the

SCHEDULE user's guide as "Working days late (according to the default calendar)." Field 7 was not provided in the *DISASTER*™ *.SD3 files, so the equivalent number was derived by subtracting the due-date from the completion date. In the case of *DISASTER*™ and TGS, lateness information was an integer value, and unused portions of the completion day were ignored in calculating these performance measures. Since no work hours were available on Saturdays or Sundays, the performance measure calculations ignored any late-time accrued on weekends. For many problems, data for only nine or eight batches were written in the *.SD3 file. This stemmed from the fact that jobs with like due-dates were consolidated into a single batch, with the batch's total quantity increased accordingly (James and Mediate, 1993:20). This circumstance made it necessary for all performance measures except Maximum Tardy Days to be weighted according to the extent of consolidation so that all ten original jobs were represented in the totals and averages. With all four performance measures, lower values are better, higher values are worse.

1. **Maximum Tardy Days (MTD).** The greatest single value in *.SD3, field 7, was selected as the MTD for each benchmark problem.

2. **Percentage of Tardy Jobs (%TJ).** Calculated by summing the number of late flags in *.SD3, field 5, dividing by the number of jobs (10), and multiplying by 100. In problems with less than ten batches, late flags for larger batches were weighted accordingly by multiplying by field 4, then dividing by the quantity of parts per job order (100). Since there were only 10 job orders, a difference of one late job resulted in a 10 percent swing in this performance measurement.

3. **Total Days Late (TDL).** Determined by multiplying field 4 (quantity) times field 7 (days late) for each batch, dividing by the quantity of parts per job order (100), then summing the batches. Multiplying by field 4 and dividing by 100 caused consolidated batches to be weighted appropriately.

4. **Average Flow Time (AFT).** Since all jobs were available for work on the first day of the planning horizon, AFT was calculated by counting the work days from the start of the planning horizon (October 3, 1993) until job completion. Totals for each batch were calculated, then multiplied by field 4 (quantity) divided by the quantity of parts per job order (100). The results were summed, and divided by the number of jobs (10), to obtain the average for the entire group of job orders.

The technique used to calculate Total Days Late (TDL) for each group of 10 job orders contained in the benchmark problems differed from the technique used by James and Mediate. The result is that the findings reported here deviate from the TDL values reported in thesis AFIT/GSM/LAS/93S-9.

Performance Categories

The researcher developed a set of categories as a means of classifying the overall performance of each software type on each benchmark problem. Considering each performance measurement of each benchmark problem, the researcher annotated the number of measurements where TGS was best, *DISASTER*™ was best, or the two software packages tied. This data could fit only 1 of 15 possible categories, labeled a through o. The categories were labeled so that category a indicated a strong performance by TGS, while category o indicated a strong performance by *DISASTER*™, with intermediate levels between the two extremes. Categories a, b, c, and d can be considered evidence of superior performance by TGS, because in these categories TGS beats or ties *DISASTER*™ in all four performance measurements. The converse is true of categories l, m, n, and o. Categories e, f, j, and k can be considered a tradeoff zone, where both software packages have good and bad performance measures. Finally,

categories g, h, and i are absolute ties. The category labels and corresponding performance aspects are listed in Table 2.

Table 2: Performance Categories

Number of better TGS performance measures	Number of equal performance measures	Number of better DISASTER performance measures	Category Label
4	0	0	a
3	1	0	b
2	2	0	c
1	3	0	d
3	0	1	e
2	1	1	f
2	0	2	g
0	4	0	h
1	2	1	i
1	1	2	j
1	0	3	k
0	3	1	l
0	2	2	m
0	1	3	n
0	0	4	o

Percent Delta Performance Measure

As a quick comparative representation of schedule performance across all four performance measures, the researcher used a percent delta measurement to indicate the percentage better or worse TGS was relative to the *DISASTER*™ measurements. A value of zero indicates a tie, a negative number indicates TGS did worse than *DISASTER*™, and a positive number indicates TGS did better than *DISASTER*™. The value was obtained by subtracting the TGS value from the *DISASTER*™ value and dividing the

result by the *DISASTER*™ value. Percent delta is calculated for each of the performance measures.

Predictions

Since a simultaneous algorithm is not restricted by fixed schedules built during earlier iterations, it stands to reason that the quality of the schedules it builds will be greater. The expected exception is the case where the sequential method identifies and schedules a single constraint resource. In such cases, the sequential algorithm is solving a simpler problem, and can be expected to perform better.

Summary

Thus far, the author has presented the need for, and background of, the research effort. This chapter described the methods used by the researcher to create schedules for the benchmark problems, using the TGS simultaneous algorithm. Details concerning the sources of data and computation of performance measures were also provided. Necessary deviations from the methods used by earlier researchers were discussed. The next chapter will restate the investigative questions and present the research findings which answer those questions.

IV. Findings

Introduction

The findings reported in this chapter pertain to the three investigative questions that this study sought to answer.

1. What in particular has changed in the algorithm between *DISASTER*™ and TGS version 2.2?

2. In terms of maximum tardiness performance, how, if at all, has the quality of schedules produced by TGS software changed with respect to solutions created by *DISASTER*™ and in relation to optimal solutions?

3. How do *DISASTER*™ and *THE GOAL SYSTEM*™ version 2.2 compare in terms of alternate performance criteria such as total days late, percentage of tardy job orders, and average flow time?

Algorithm Differences

At the macro level, TGS software performs much like its older sibling, *DISASTER*™. The operation remains faithful to the procedures described in *The Haystack Syndrome*. The sequence of identifying constraint(s), building and leveling the ruins, and subordinating non-constraint resources to the drum remains the same. There is now an opportunity to deviate from the iterative process of sequentially identifying and scheduling constraints one at a time. Multiple resources can be identified by the operator for drum creation during the first (and only) iteration. This feature can be a double-edged sword. As long as the constraints chosen by the operator to be simultaneously scheduled would have been identified anyway, the simultaneous algorithm saves the effort of looping through the ruins-building and leveling process several times. However, when

building drums sequentially it is not uncommon for one or more seemingly interactive constraints to drop out following the first or subsequent iterations. In such cases, drums will be built by the simultaneous algorithm that would not have been built by the sequential algorithm. In fact, the SCHEDULE user's guide is conspicuously passive on the subject. The only method discussed for accessing the simultaneous algorithm is during the resolution of a drum violation following sequential drum building. There is only a hint in the user's guide that the simultaneous algorithm even exists. It can be found on page 11-7 under the discussion of three types of drum loop that can be made. The simultaneous algorithm is touted as the most effective of the drum loops available. However, it appears that the manufacturer's expectation is that the simultaneous algorithm will be used only in the *resolution of a drum violation*, not as a first choice in *drum building*.

Performance Differences With Respect to Optimal Solutions

For the 108 benchmark scheduling problems used in this study, the only measure of performance with optimal solutions available was Maximum Tardy Days (MTD). In fact, only 84 of the benchmark solutions have been successfully solved by Carlson and Lettiere's branch and bound method. This paragraph concentrates only on the 84 problems for which optimal solutions with respect to MTD are known. *DISASTER*™ only performed worse than the optimal solution in 10 cases. TGS performed slightly better, performing worse than the optimal solution in only 4 cases. *DISASTER*™ and TGS tied in 72 of the 84 cases, 69 of those being optimal solutions. In 10 of the 12

remaining cases, the TGS solution resulted in a lower (better) MTD figure than the best of the two *DISASTER*™ solutions. Solutions that differed, only did so by one day. Surprisingly, in three cases the TGS solution actually yielded MTD values lower than the optimums reported by Carlson and Lettiere. Feasibility of the TGS solutions has been verified by the researcher. Output products for one of these problems (T 125 25 R2) are presented in greater detail in Appendix C.

Performance with Respect to Other Measures

This section reveals differences between the simultaneous and sequential algorithms over a broader range of performance measures. This includes a look at the MTD measurements of cases excluded from the previous paragraph's discussion. In general, the simultaneous algorithm turned in a much weaker performance than anticipated. An exhaustive list of measured performance characteristics can be found in Appendix D. Appendix E highlights which algorithm performed best in each performance measure, for each benchmark problem. It also lists the percentage of difference in performance measures between TGS and *DISASTER*™. The present section contains distilled data in tabular and graphical form for ease of comprehension.

Table 3 summarizes the descriptive statistics for *DISASTER*™ best, *DISASTER*™ blue first, *DISASTER*™ gold first, and *THE GOAL SYSTEM*™.

Table 3: Descriptive Statistics for Performance Measures

		MTD	%TJ	TDL	AFT
$DISASTER_{Blue}$	Mean	9.35	89.54	48.92	10.41
	Median	9.00	90.00	46.50	10.00
	Std Dev	3.90	12.33	19.43	1.82
$DISASTER_{Gold}$	Mean	7.43	88.06	40.88	9.61
	Median	7.00	90.00	38.50	9.50
	Std Dev	2.77	13.29	15.53	1.26
$DISASTER_{Best}$	Mean	7.27	85.56	40.37	9.55
	Median	7.00	90.00	38.00	9.50
	Std Dev	2.81	13.90	15.54	1.27
TGS 2.2	Mean	7.19	92.31	42.91	9.86
	Median	7.00	100.00	40.50	9.70
	Std Dev	2.84	12.80	16.94	1.35

The statistics indicated that *DISASTER*™ generally performed better when the gold resource was identified as the primary constraint. The median values with gold as the primary constraint were virtually identical to the overall best performances. In the 108 benchmark scheduling problems the gold resource was always the most heavily loaded resource. There was some improvement in the means for $DISASTER_{Best}$ when we picked the strongest performance achieved on a case-by-case basis. Remember, the choice of best performer was not a selection of the single best schedule for each benchmark problem, but the minimum value for each measure per each benchmark problem. This is somewhat unrealistic when considering that in a real environment the scheduler would have to choose one schedule or the other--not accept bits and pieces of

each schedule. However, in a real environment the scheduler would not try to (or be able to) optimize all performance indicators simultaneously.

Based on the mean values, the TGS simultaneous algorithm appeared to have a slight edge over the *DISASTER*™ algorithm in terms of minimizing MTD. But the substantial worsening in the other performance measures indicated a general reduction in schedule quality when the simultaneous algorithm was used. The mean, median, and standard deviations of the other performance measures all indicated a worse performance.

As for the frequency with which an algorithm performed best with respect to a particular performance measure, TGS fell short of *DISASTER*™'s performance again. The left side of Table 4 is a tally of the number of times an algorithm performed best as measured by a particular performance measure. The right side makes the same sort of comparison, using *DISASTER*™ gold-first and blue-first as the objects of comparison. While the blue-first sequence rarely scored best in MTD, it achieved much better standing among the other performance measures.

Table 4: Best Performance by Algorithm and Constraint Sequence

	MTD	%TJ	TDL	AFT			MTD	%TJ	TDL	AFT
D_{best}	5	62	87	93		gold	77	34	80	77
tied	89	45	14	10		tied	25	50	5	6
TGS	14	1	7	5		blue	6	24	23	25
total	108	108	108	108		total	108	108	108	108

Counting the number of best performances by each algorithm in each benchmark problem confirmed the poorer performance of the simultaneous algorithm. The results are listed in Table 5. In 82 of 108 cases, the sequential algorithm performed better than or equal to the simultaneous algorithm in all four performance measurements. In stark contrast, the reverse was true in only 2 cases. The results can be seen graphically in Figure 6. James and Mediate learned that when building drums sequentially, the secondary constraint was occasionally judged during subordination to be capable of supporting the pace of the primary drum, making a second drum unnecessary. Considering that these single drum solutions might skew the results in favor of the sequential algorithm, the researcher eliminated all cases where *DISASTER*™ was able to create a single-drum solution. As seen in Table 5 and Figure 7, the effects of filtering the data in this way did not eliminate the overwhelming performance differences between the two algorithms.

Table 5: Categorization of Benchmark Problems by Algorithm Performance

Number of better TGS performance measures	Number of equal performance measures	Number of better DISASTER performance measures	Category Label	Occurrences among all 108 benchmark problems	Occurrences among 69 interactive problems
4	0	0	a	0	0
3	1	0	b	1	1
2	2	0	c	1	1
1	3	0	d	0	0
3	0	1	e	1	1
2	1	1	f	2	2
2	0	2	g	0	0
0	4	0	h	6	2
1	2	1	i	2	2
1	1	2	j	2	2
1	0	3	k	11	11
0	3	1	l	3	3
0	2	2	m	38	15
0	1	3	n	38	29
0	0	4	o	3	0
			total ➔	108	69

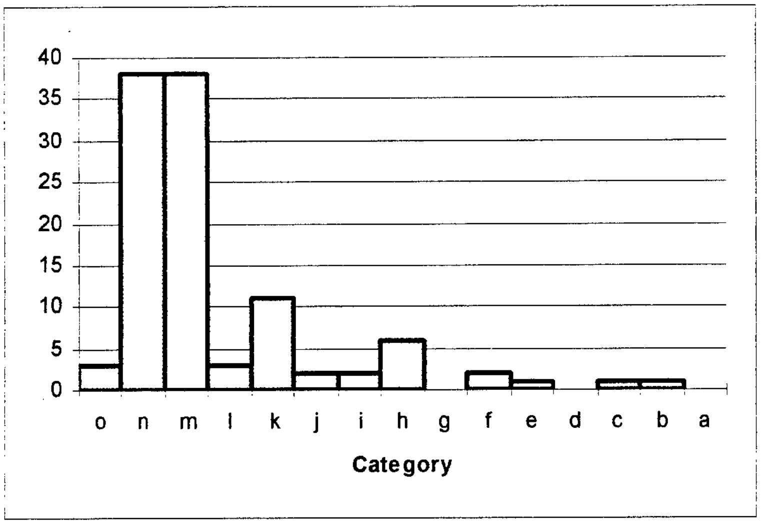

Figure 6: Benchmark Problems Categorized by Algorithm Performance

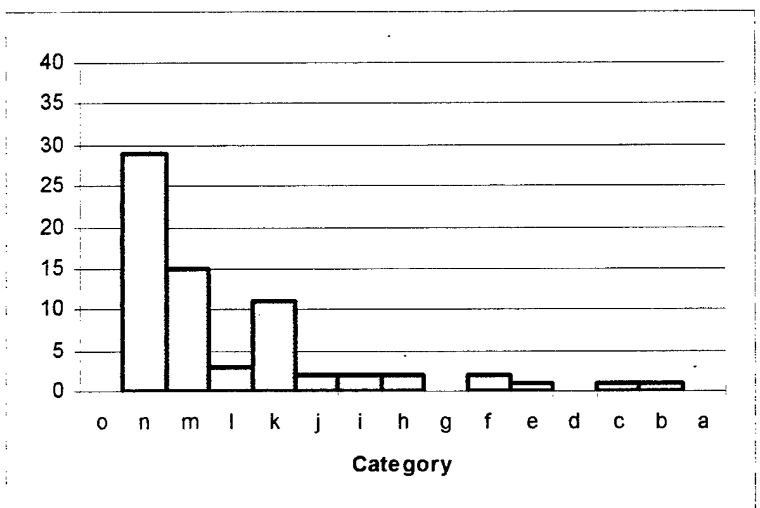

Figure 7: Interactive Problems Categorized by Algorithm Performance

A histogram in Figure 8 illustrates a high degree of similarity between the performance of the algorithms in terms of performance measure MTD. Conversely,

examination of the histograms in Figures 9, 10, and 11 reveals the dissimilarities between the performances of the two algorithms on the other three performance measures.

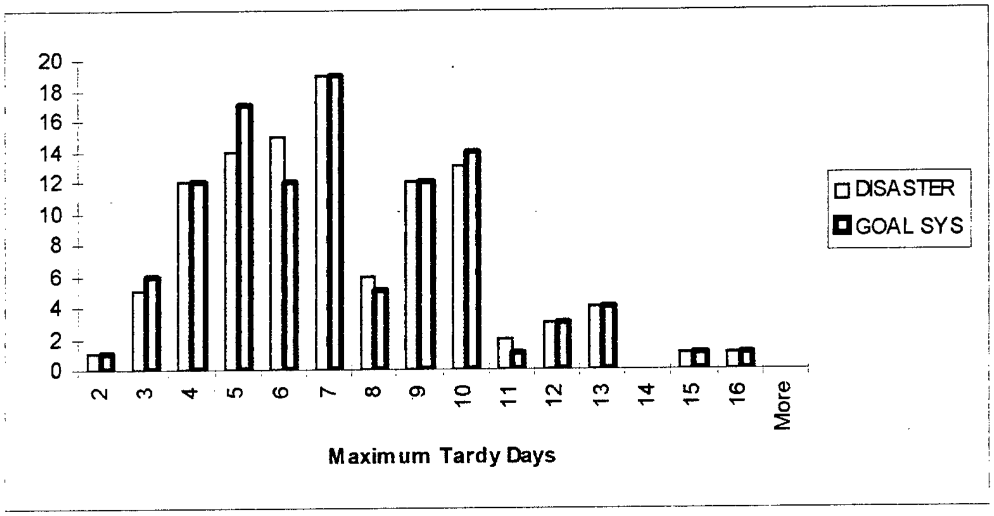

Figure 8: Histogram of MTD

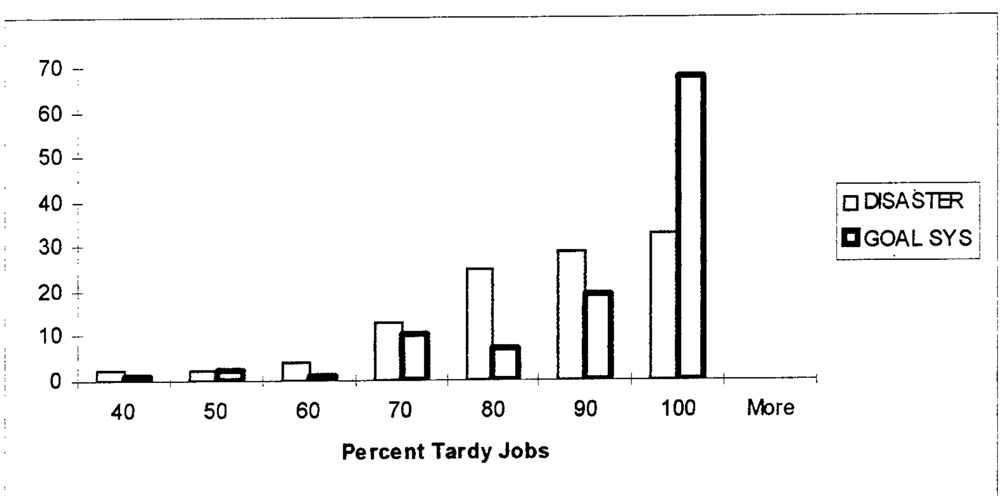

Figure 9: Histogram of %TJ

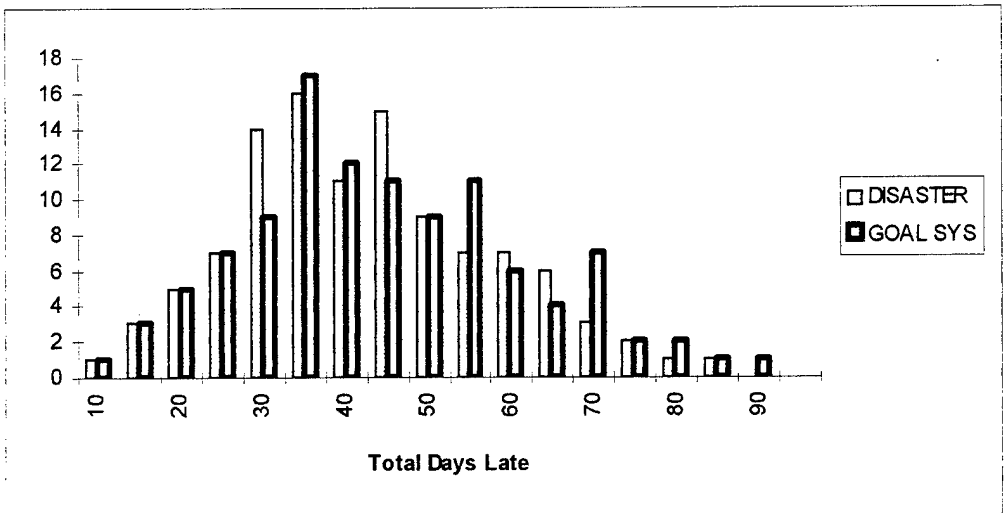

Figure 10: Histogram of TDL

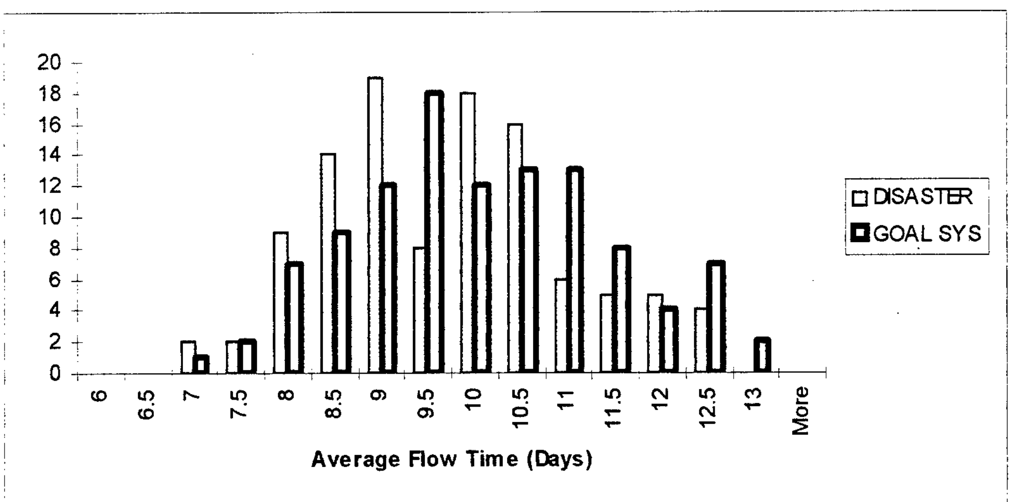

Figure 11: Histogram of AFT

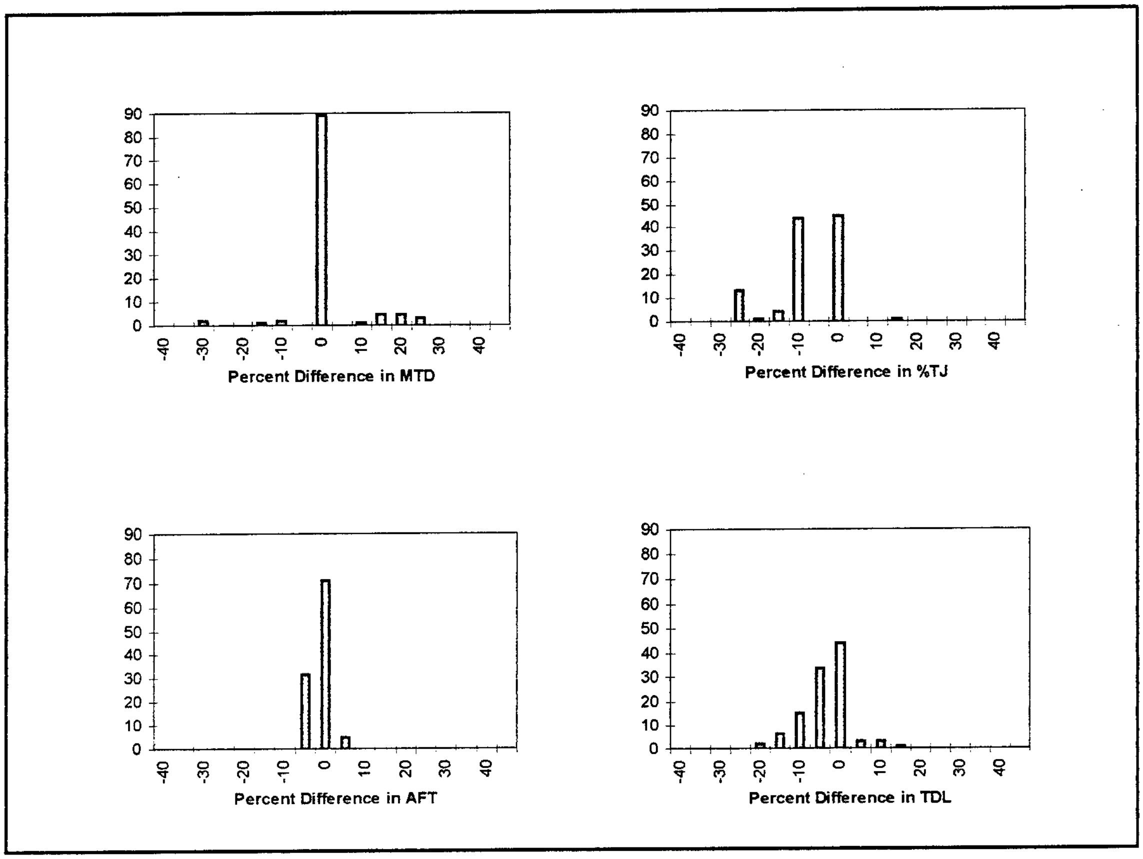

Figure 12: Histograms of %Δ in Performance Measures

As described in Chapter III, the percent delta measurement represents the percentage better or worse TGS performed than *DISASTER*™, and how often. MTD values were roughly equivalent, but the other measures showed a definite negative trend. The most significant negative differences in %Δ MTD and %TJ can be explained by the non-interactive constraint problems that the sequential algorithm solved with a single drum. The data used to derive these histograms is available in Appendix D.

Summary

This chapter presented the findings of the research as they apply to the investigative questions. Macro-level differences between the *DISASTER*™ and *THE GOAL SYSTEM*™ algorithms were discussed. Graphical and tabular representations of performance data were then presented and discussed. The next chapter contains conclusions based on the results of the research and recommendations for future study.

V. Conclusions

Introduction

This section provides a summary of the research effort. The investigative questions are addressed, including the author's interpretation of the findings. Suggestions for further research are provided.

Summary of Thesis

THE GOAL SYSTEM™ (TGS) is the latest release of a family of Drum-Buffer-Rope scheduling software which includes Optimized Production Technology (OPT) and *DISASTER*™. A criticism of earlier versions of the software was the sequential nature of its algorithm. Indeed, a weakness of the sequential algorithm has been demonstrated in that the sequence in which multiple interactive constraints are identified and exploited has a significant effect on the quality of schedules produced. With the release of *THE GOAL SYSTEM*™ version 2.2, a simultaneous algorithm was made available. In order to test the hypothesis that that the simultaneous algorithm would produce a higher quality schedule, this research sought to answer three questions.

1. What in particular has changed in the algorithm between *DISASTER*™ and TGS version 2.2?

To answer this question, the researcher examined the user's guide for the TGS SCHEDULE software module. The documentation provided few clues of the very existence of a simultaneous scheduling algorithm in TGS. Certainly it was not billed as a revolutionary improvement or a major conceptual shift. The user's guide described an

iterative process to schedule-building, with little conceptual distinction from the *DISASTER*™ method. The researcher next turned to observing the operation of the software in hopes of uncovering differences from earlier descriptions of *DISASTER*™ operations. At the macro-level, the two packages operated in essentially the same manner. The one noticeable characteristic of TGS was that when constraints were scheduled simultaneously, the operator was never confronted with a drum violation.

2. In terms of maximum tardiness performance, how, if at all, has the quality of schedules produced by TGS software changed with respect to solutions created by *DISASTER*™ and in relation to optimal solutions?

TGS performed better when measured by MTD than by any other performance measure. The mean value of the TGS solutions was better than that of the *DISASTER*™ solutions. There were only five cases in which TGS did not beat or tie the *DISASTER*™ solution. Three of these cases were solved by *DISASTER*™ by scheduling only one constraint, so that it essentially solved an easier problem. In all but two cases, the maximum difference between optimal, TGS, and *DISASTER*™ was only one day. In three cases, the TGS solution was better than the optimal solution. Output products for one of these cases are re-created in Appendix C, should the reader wish a more detailed examination.

3. How do *DISASTER*™ and *THE GOAL SYSTEM*™ version 2.2 compare in terms of alternate performance criteria such as total days late, percentage of tardy job orders, and average flow time?

The additional performance measures selected for comparing the two algorithms were percentage of tardy jobs (%TJ), total days late (TDL), and average flow-time (AFT).

The first two provided additional due-date performance measures. AFT provided an indication of cost in that the relationship between work-in-process inventory and AFT is very close. TGS did not perform as well when judged by these performance measures as it did when judged by the MTD measurement. TGS's mean values for each of these performance measures were worse than those of *DISASTER*™, and seldom did TGS beat *DISASTER*™ in any of the performance measures. *DISASTER*™ outperformed TGS in 57 percent of cases as measured by %TJ, 81 percent of cases as measured by TDL, and 86 percent of cases as measured by AFT.

The hypothesis that TGS version 2.2 scheduling software, using a simultaneous scheduling algorithm, produces a higher quality schedule for production scenarios involving interactive multiple constraints than the earlier *DISASTER*™ software, using a sequential algorithm, is not supported by the results of this research.

It is the conclusion of the researcher that, based on all performance measures considered, the simultaneous algorithm used in TGS version 2.2 is not suited for building drum schedules that exceed, or even equal the quality of those created by *DISASTER*™. The simultaneous algorithm is easier to use than the sequential one, so there is some value in that. The researcher speculates that TGS was intended to be used primarily as a sequential scheduler, with the simultaneous feature only meant for use in resolving drum violations.

Future Research

This study has indicated that the best of the *DISASTER*™ solutions to the benchmark problems are generally better than the solutions obtained by using the simultaneous solution algorithm of *THE GOAL SYSTEM*™ version 2.2. An opportunity for further research would be a similar comparison based on real world scheduling situations rather than Captains James and Mediate's benchmark problems.

There are a great many finite capacity scheduling software packages on the market today. A comparison between *THE GOAL SYSTEM*™ and one or more programs outside of the Theory of Constraints family would examine the effectiveness of the Drum-Buffer-Rope (DBR) methodology in contrast to alternate techniques.

Although there was some interest in using *DISASTER*™ at Air Force Air Logistics Centers (ALC) a number of years ago, there does not seem to be great interest in doing so at this time. A survey of ALC operations may reveal that their situations are not suited to use of *THE GOAL SYSTEM*™, or that some other incompatibility exists which has prevented its continued use, although Guide and Ghiselli have described a successful DBR implementation at the Alameda Naval Aviation Depot (1995:79-83).

Finally, a study of *THE GOAL SYSTEM* ™ algorithm at the micro-level may reveal why the simultaneous algorithm in this study produced such markedly worse performance than expected.

Appendix A: Glossary

Benchmark problems: A set of 108 scheduling situations created by Captains James and Mediate to represent diverse scenarios, intended for use in comparing the performance of scheduling methods

Buffer: According to Theory of Constraints philosophy, a period of time used to decouple an event from statistical fluctuations in a preceding event

Capacity: Maximum quantity of work or output a resource can produce over a given time period

Capacity constraint resource: A resource which has insufficient capacity, or has sufficient raw capacity but becomes a constraint anyway through improper scheduling

Completion date: The date on which the processing of a job's final operation is completed

Constraint: An element which restricts the performance of the organization because we don't have enough of it

Demand: Requirement for the work or output produced by a resource

DISASTER™: A Drum-Buffer-Rope scheduling software package which sequentially identifies and exploits constraints

Drum: According to Theory of Constraints philosophy, the schedule for a constraint resource which sets the pace for the system

Drum-Buffer-Rope: A scheduling method that coheres to the Theory of Constraints philosophy by only building schedules for constraints, using buffers to protect throughput of the system, and regulating production by non-constraints through metered input of raw materials

Drum Building: The process of building a finite capacity schedule for a resource

Drum Violation: Sometimes occurs during drum building, when batches can not be placed due to timing conflicts with earlier drums

Due-date: The time at which an external agency desires the job to leave the shop, therefore, the time by which all operations should be complete

Exploit: Action taken to ensure maximum utility is gained from a constraint resource

First Day Load (FDL) peak: A situation where the demand for a non-constraint resource on the first day of the planning horizon exceeds the resource's capacity

Flow time: The total time that the job spends in the shop, whether idle or being processed

Horizon: The time period of concern when building a schedule

Interactive Constraint: A constraint which feeds or is fed by another constraint

Job Order: A quantity of like product types, demanded for a single due-date, by a single customer

Lateness: A measure of performance, the algebraic difference between due-date and completion date

Measures of performance: means for classifying the degree to which a schedule meets the desired objectives of the system

Non-Constraint Resource: A resource which has sufficient capacity to meet demand

Plant Type: Classification of a manufacturing operation by its dominant resource/product interaction characteristics; divergent, convergent, and assemble-to-order

Primary Constraint: The first constraint to be scheduled, with secondary constraints subject to the timing it imposes

Product Type: A finished good that requires a unique combination of resources in its creation

Resource: A factor of production, like a tool, a machine, or an operator

Rope: The schedule for introducing raw materials into the system–An indirect schedule for non-constraint resources

Ruins: A Gantt chart representing an infinite capacity schedule

Secondary Constraint: A constraint to be scheduled subject to the timing imposed by the schedule of a primary constraint

Sequential drum building: An iterative procedure for building schedules for two or more constraints, where the schedule for the secondary constraint is subject to timing imposed by the schedule for the primary constraint

Simultaneous drum building: A procedure for creating schedules for two or more constraint resources without granting either one primacy

Subordinate: Controlling the output of a non-constraint resource so that it meets, but does not exceed the pace set by the drum

Tardiness: A measure of due-date performance similar to lateness, but never less than zero

Work-in-process inventory: Materials which have begun processing through the system, but have not yet completed the final operation–Paid for, but not available for sale

Appendix B: Benchmark Scheduling Problem Plant Layouts
(adapted from James and Mediate, 1993)

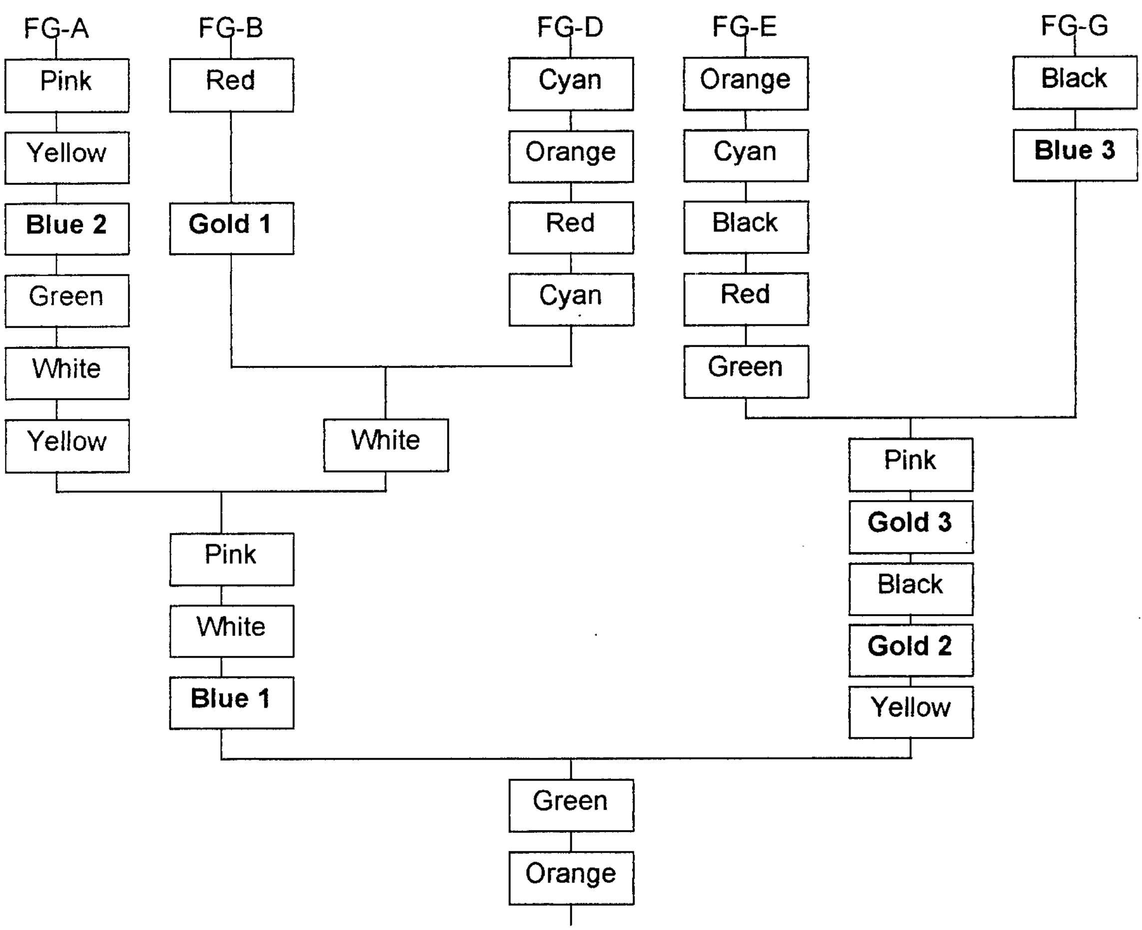

Figure 13: V-Plant

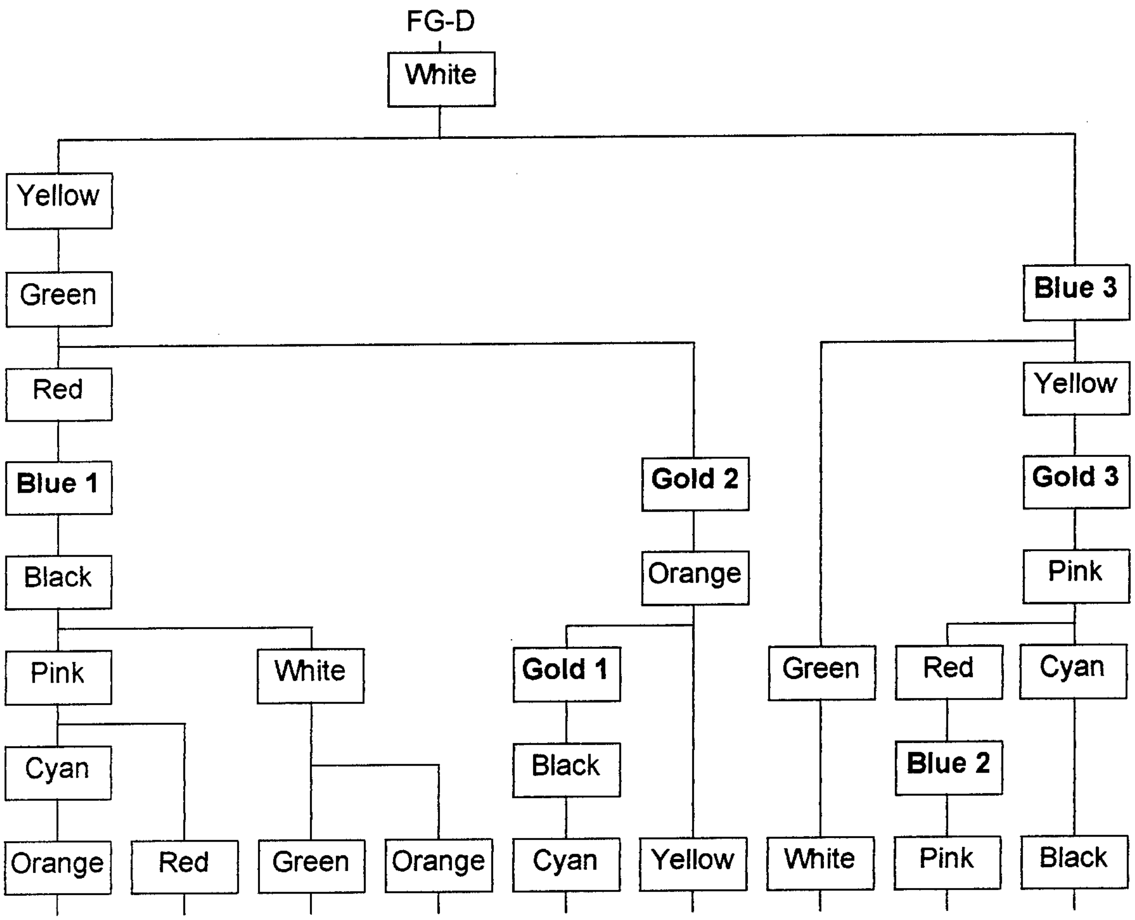

Figure 14: A-Plant

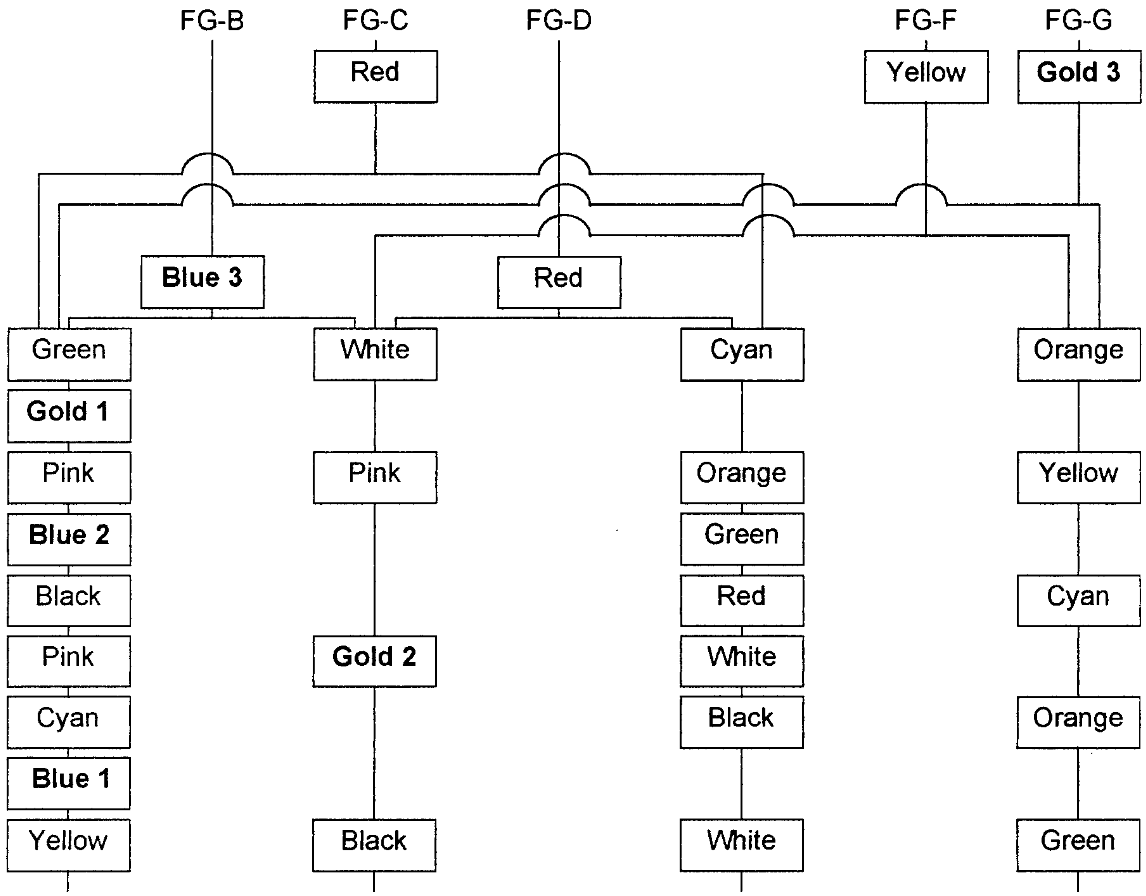

Figure 15: T-Plant

Appendix C: Output Products for Sample Problem

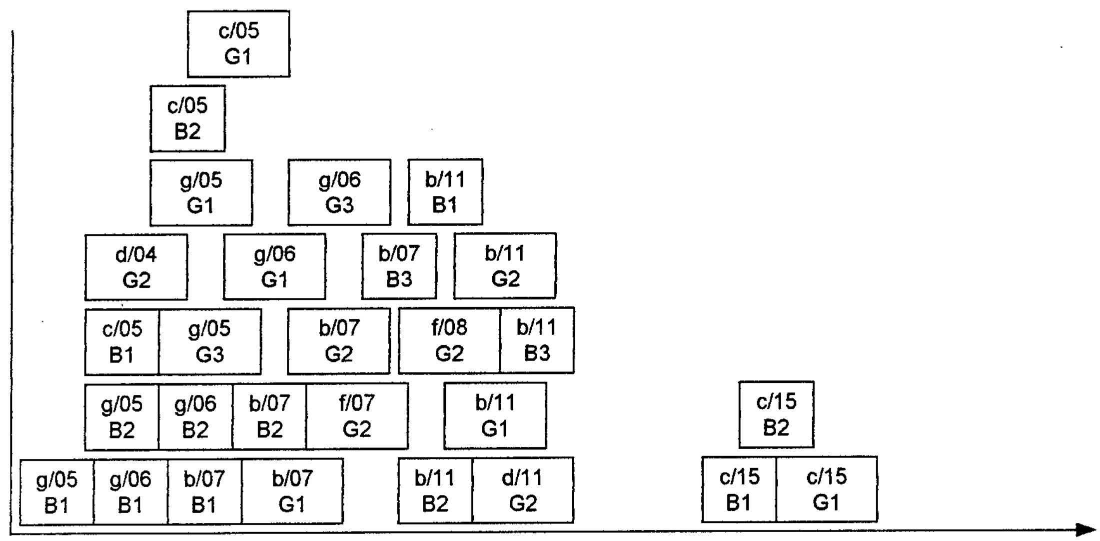

Figure 16: Ruins, Benchmark Problem T 125 25 R2

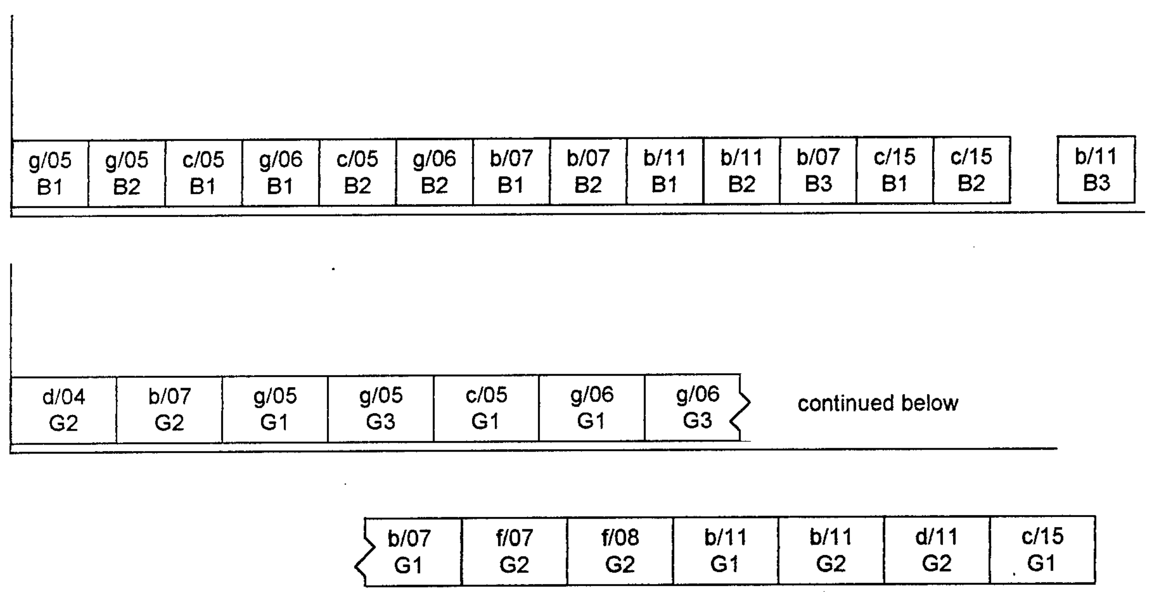

Figure 17: Drums, Benchmark Problem T 125 25 R2

Key to designations:

Top Line–product type/due date

Bottom Line–Constraint Station (see plant layout)

Table 6: Drum Schedule, T 125 25 R2

Resource Type	Station	Product Type/Job Number	Qty	Processing Time	Ideal Stop Date	Ideal Stop Time	Start Date	Start Time	Stop Date	Stop Time
gold	G2	b-fg/931007	100	5.4	931006	3:54	931005	1:00	931006	2:00
blue	B1	b-fg/931007	100	4.3	931005	1:54	931011	3:00	931012	2:10
blue	B2	b-fg/931007	100	4.3	931005	5:59	931012	2:10	931013	1:20
gold	G1	b-fg/931007	100	5.4	931006	3:54	931013	7:00	931015	0:00
blue	B3	b-fg/931007	100	4.3	931006	7:59	931014	7:40	931015	6:50
blue	B1	b-fg/931011	100	4.3	931007	1:54	931013	1:20	931014	0:30
blue	B2	b-fg/931011	100	4.3	931007	5:59	931014	0:30	931014	7:40
gold	G1	b-fg/931011	100	5.4	931008	3:54	931019	2:00	931020	3:00
gold	G2	b-fg/931011	100	5.4	931008	3:54	931020	3:00	931021	4:00
blue	B3	b-fg/931011	100	4.3	931008	7:59	931021	0:55	931022	0:05
blue	B1	c-fg/931005	100	4.3	931001	5:59	931005	6:20	931006	5:30
blue	B2	c-fg/931005	100	4.3	931004	2:04	931007	4:40	931008	3:50
gold	G1	c-fg/931005	100	5.4	931004	7:59	931008	4:00	931011	5:00
blue	B1	c-fg/931015	100	4.3	931013	5:59	931015	6:50	931018	6:00
blue	B2	c-fg/931015	100	4.3	931014	2:04	931018	6:00	931019	5:10
gold	G1	c-fg/931015	100	5.4	931014	7:59	931022	5:00	931025	6:00
gold	G2	d-fg/931004	100	5.4	931001	7:59	931003	0:00	931005	1:00
gold	G2	d-fg/931011	100	5.4	931008	7:59	931021	4:00	931022	5:00
gold	G2	f-fg/931007	100	5.4	931006	7:59	931015	0:00	931018	1:00
gold	G2	f-fg/931008	100	5.4	931007	7:59	931018	1:00	931019	2:00
blue	B1	g-fg/931005	100	4.3	931001	1:53	931003	0:00	931004	7:10
blue	B2	g-fg/931005	100	4.3	931001	5:58	931004	7:10	931005	6:20
gold	G1	g-fg/931005	100	5.4	931004	3:53	931006	2:00	931007	3:00
gold	G3	g-fg/931005	100	5.4	931004	7:59	931007	3:00	931008	4:00
blue	B1	g-fg/931006	100	4.3	931004	1:53	931006	5:30	931007	4:40
blue	B2	g-fg/931006	100	4.3	931004	5:58	931008	3:50	931011	3:00
gold	G1	g-fg/931006	100	5.4	931005	3:53	931011	5:00	931012	6:00
gold	G3	g-fg/931006	100	5.4	931005	7:59	931012	6:00	931013	7:00

sorted by product type / job order number

Appendix D: Performance Measurement Data

Plant type	% RCF	% delta RCF	Rep	ALL CASE	Optimal	THE GOAL SYSTEM version 2.2				DISASTER best				Dual con-straints
				MTD	(or NO)	MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	(or NO)
V	105	0	1	5		5	80	29	8.7	5	70	26	8.4	
V	105	0	2	4		4	100	26	8.1	4	90	25	8	
V	105	0	3	3		4	70	18	8.2	3	80	20	8.4	
V	105	0	4	8		8	100	52	8.9	8	90	47	8.4	
V	105	25	1	8		8	70	37	9.5	8	70	36	9.4	
V	105	25	2	5		5	90	31	8.6	6	80	29	8.4	
V	105	25	3	5		5	80	27	9.1	5	80	26	9	
V	105	25	4	11		11	100	60	9.7	11	90	56	9.3	
V	105	50	1	10		10	70	47	10.5	10	70	43	10	
V	105	50	2	7		7	90	39	9.4	8	70	34	8.9	
V	105	50	3	7		7	80	34	9.8	7	80	32	9.6	
V	105	50	4	13		13	100	71	10.8	13	90	65	10	
V	115	0	1	6		6	90	35	9.3	6	80	31	8.9	
V	115	0	2	5		5	100	31	8.6	5	90	30	8.5	
V	115	0	3	4		4	90	25	8.9	4	80	26	9	
V	115	0	4	9		9	100	58	9.5	9	90	53	9	
V	115	25	1	9		9	90	45	10.3	9	70	38	9.6	
V	115	25	2	6		6	100	38	9.3	7	80	35	9	
V	115	25	3	6		6	90	34	9.8	6	80	37	9.7	
V	115	25	4	12		12	100	67	10.4	12	90	61	9.8	
V	115	50	1	12		12	90	57	11.5	12	70	49	10.7	
V	115	50	2	9		9	100	47	10.2	9	80	43	9.8	
V	115	50	3	9		9	90	44	10.8	9	80	41	10.5	
V	115	50	4	15		15	100	80	11.7	15	90	72	10.8	
V	125	0	1	7		7	90	43	10.1	7	90	40	9.8	
V	125	0	2	6		6	100	37	9.2	6	90	34	8.9	
V	125	0	3	4		5	100	32	9.6	5	90	33	9.7	
V	125	0	4	10		10	100	65	10.2	10	100	60	9.7	
V	125	25	1	10		10	90	53	11.1	10	90	46	10.4	
V	125	25	2	7		7	100	45	10	8	90	41	9.6	
V	125	25	3	7		7	100	41	10.5	7	90	43	10.3	
V	125	25	4	13		13	100	75	11.2	13	100	69	10.6	
V	125	50	1	13		13	100	67	12.5	13	90	60	11.7	
V	125	50	2	10		10	100	55	11	10	90	49	10.4	
V	125	50	3	10		10	100	52	11.6	10	90	50	11.4	
V	125	50	4	6		16	100	89	12.6	16	100	81	11.7	

Plant type	% RCF.	% delta RCF	Rep	ALL CASE	Optimal	THE GOAL SYSTEM version 2.2				DISASTER best				Dual con-straints
				MTD	(or NO)	MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	(or NO)
A	105	0	1	4	NO	3	100	26	7.9	4	100	29	8.2	
A	105	0	2	2	NO	3	80	14	8.1	3	80	15	8.2	
A	105	0	3	3	NO	2	100	17	7.8	2	90	17	7.8	
A	105	0	4	2	NO	3	100	24	7.8	4	90	24	7.8	
A	105	25	1	5		5	100	40	9.3	5	100	38	9.1	NO
A	105	25	2	5		5	100	29	9.6	5	100	26	9.3	NO
A	105	25	3	5		5	100	31	9.2	5	100	28	8.9	NO
A	105	25	4	5		5	100	37	9.1	5	100	36	9	NO
A	105	50	1	7		7	100	52	10.5	7	100	51	10.4	NO
A	105	50	2	7		7	100	40	10.7	7	100	38	10.5	NO
A	105	50	3	7		7	100	45	10.6	7	100	44	10.5	NO
A	105	50	4	7		7	100	52	10.6	7	100	51	10.5	NO
A	115	0	1	4	NO	4	100	33	8.6	4	100	32	8.5	
A	115	0	2	3		4	90	21	8.8	3	80	21	8.8	
A	115	0	3	3	NO	3	100	24	8.5	3	100	22	8.3	
A	115	0	4	3	NO	4	100	33	8.7	4	90	27	8.1	
A	115	25	1	6	NO	6	100	46	9.9	6	100	44	9.7	NO
A	115	25	2	6		6	100	35	10.2	6	100	32	9.9	NO
A	115	25	3	6		6	100	35	9.6	6	100	35	9.6	NO
A	115	25	4	6		6	100	44	9.8	6	100	42	9.6	NO
A	115	50	1	9		9	100	60	11.3	9	100	60	11.3	NO
A	115	50	2	9		9	100	49	11.6	9	100	48	11.5	NO
A	115	50	3	9		9	100	54	11.5	9	100	54	11.5	NO
A	115	50	4	9		9	100	60	11.4	9	100	59	11.3	NO
A	125	0	1	4	NO	4	100	40	9.3	4	100	37	9	
A	125	0	2	4		4	100	27	9.4	4	90	26	9.3	
A	125	0	3	4	NO	4	100	29	9	4	100	29	9	
A	125	0	4	5	NO	5	100	37	9.1	5	100	36	9	
A	125	25	1	7		7	100	52	10.5	7	100	51	10.4	NO
A	125	25	2	7		7	100	40	10.7	7	100	38	10.5	NO
A	125	25	3	7		7	100	45	10.6	7	100	44	10.5	NO
A	125	25	4	7		7	100	52	10.6	7	100	51	10.5	NO
A	125	50	1	10		10	100	70	12.3	10	100	69	12.2	NO
A	125	50	2	10		10	100	58	12.5	10	100	57	12.4	NO
A	125	50	3	10		10	100	63	12.4	10	100	63	12.4	NO
A	125	50	4	10		10	100	70	12.4	10	100	68	12.2	NO

Plant type	% RCF	% delta RCF	Rep	ALL CASE	Optimal	THE GOAL SYSTEM version 2.2				DISASTER best				Dual con-straints
				MTD	(or NO)	MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	(or NO)
T	105	0	1	4		4	70	22	7.7	4	70	22	7.7	
T	105	0	2	5		5	90	35	7.8	6	80	35	7.8	
T	105	0	3	3	NO	3	40	9	7.2	3	40	9	6.9	
T	105	0	4	3		3	50	12	7	4	40	11	6.8	
T	105	25	1	5		5	100	35	9	5	80	31	8.5	
T	105	25	2	8		8	100	51	9.4	8	80	46	8.9	
T	105	25	3	5		5	70	20	8.3	5	70	19	7.9	NO
T	105	25	4	5	NO	5	80	24	8.2	6	70	23	8	
T	105	50	1	7		7	100	50	10.5	7	90	45	9.9	
T	105	50	2	10		10	100	67	11	10	80	63	10.5	NO
T	105	50	3	7		7	70	32	9.5	7	70	31	9.1	NO
T	105	50	4	8	NO	8	90	39	9.7	7	80	33	9	NO
T	115	0	1	5		5	80	29	8.4	5	70	28	8.2	
T	115	0	2	6		6	90	42	8.5	7	80	41	8.4	
T	115	0	3	4	NO	4	50	14	7.7	4	50	14	7.4	
T	115	0	4	4	NO	4	70	17	7.5	5	60	16	7.3	
T	115	25	1	6	NO	6	100	45	10	6	80	39	9.3	
T	115	25	2	9		9	100	62	10.5	9	80	57	10	NO
T	115	25	3	6		6	70	28	9.1	6	60	26	8.6	NO
T	115	25	4	7	NO	7	80	34	9.2	6	70	28	8.5	NO
T	115	50	1	9		9	100	62	11.7	9	90	55	10.9	
T	115	50	2	12		12	100	78	12.1	12	90	73	11.6	NO
T	115	50	3	9		9	90	43	10.6	9	80	43	10.3	NO
T	115	50	4	9	NO	9	100	48	10.6	9	90	44	10.1	NO
T	125	0	1	5	NO	6	90	35	9	6	70	32	8.7	
T	125	0	2	8		7	90	46	8.9	8	80	45	8.8	
T	125	0	3	5	NO	5	60	19	8.2	5	50	19	7.9	
T	125	0	4	5	NO	5	70	22	8	6	60	21	7.8	
T	125	25	1	7		7	100	50	10.5	7	90	45	9.9	
T	125	25	2	11		10	100	67	11	11	80	61	10.4	
T	125	25	3	7		7	70	32	9.5	7	60	31	9.1	NO
T	125	25	4	8	NO	8	90	39	9.7	7	80	33	9	NO
T	125	50	1	10		10	100	67	12.2	10	90	62	11.6	
T	125	50	2	14		13	100	85	12.8	13	90	78	12	NO
T	125	50	3	10		10	90	49	11.2	10	80	48	10.8	NO
T	125	50	4	10	NO	10	100	55	11.3	10	90	50	10.7	NO

Appendix E: Performance Comparison Data

Benchmark Problem				DISASTER or TGS				Percent Δ				Blue or Gold				Dual
Identification				MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	
V	105	0	1	tied	DIS	DIS	DIS	0.0	-14.29	-11.54	-3.57	gold	gold	gold	gold	
V	105	0	2	tied	DIS	DIS	DIS	0.0	-11.11	-4.0	-1.25	tied	tied	tied	tied	
V	105	0	3	DIS	TGS	TGS	TGS	-33.33	12.5	10.0	2.38	gold	blue	gold	gold	
V	105	0	4	tied	DIS	DIS	DIS	0.0	-11.11	-10.64	-5.95	tied	blue	blue	blue	
V	105	25	1	tied	tied	DIS	DIS	0.0	0.0	-2.78	-1.06	gold	gold	gold	gold	
V	105	25	2	TGS	DIS	DIS	DIS	16.67	-12.5	-6.9	-2.38	gold	gold	gold	gold	
V	105	25	3	tied	tied	DIS	DIS	0.0	0.0	-3.85	-1.11	tied	tied	gold	gold	
V	105	25	4	tied	DIS	DIS	DIS	0.0	-11.11	-7.14	-4.3	tied	gold	gold	gold	
V	105	50	1	tied	tied	DIS	DIS	0.0	0.0	-9.3	-5.0	gold	gold	gold	blue	
V	105	50	2	TGS	DIS	DIS	DIS	12.5	-28.57	-14.71	-5.62	gold	gold	gold	gold	
V	105	50	3	tied	tied	DIS	DIS	0.0	0.0	-6.25	-2.08	gold	tied	gold	gold	
V	105	50	4	tied	DIS	DIS	DIS	0.0	-11.11	-9.23	-8.0	gold	gold	tied	blue	
V	115	0	1	tied	DIS	DIS	DIS	0.0	-12.5	-12.9	-4.49	gold	gold	gold	gold	
V	115	0	2	tied	DIS	DIS	DIS	0.0	-11.11	-3.33	-1.18	tied	tied	tied	tied	
V	115	0	3	tied	DIS	TGS	TGS	0.0	-12.5	3.85	1.11	tied	blue	gold	gold	
V	115	0	4	tied	DIS	DIS	DIS	0.0	-11.11	-9.43	-5.56	tied	blue	blue	blue	
V	115	25	1	tied	DIS	DIS	DIS	0.0	-28.57	-18.42	-7.29	gold	gold	gold	gold	
V	115	25	2	TGS	DIS	DIS	DIS	14.29	-25.0	-8.57	-3.33	gold	gold	gold	gold	
V	115	25	3	tied	DIS	TGS	DIS	0.0	-12.5	8.11	-1.03	gold	blue	blue	blue	
V	115	25	4	tied	DIS	DIS	DIS	0.0	-11.11	-9.84	-6.12	gold	gold	gold	gold	
V	115	50	1	tied	DIS	DIS	DIS	0.0	-28.57	-16.33	-7.48	gold	gold	gold	gold	
V	115	50	2	tied	DIS	DIS	DIS	0.0	-25.0	-9.3	-4.08	gold	gold	gold	gold	
V	115	50	3	tied	DIS	DIS	DIS	0.0	-12.5	-7.32	-2.86	gold	tied	gold	gold	
V	115	50	4	tied	DIS	DIS	DIS	0.0	-11.11	-11.11	-8.33	gold	gold	gold	blue	
V	125	0	1	tied	tied	DIS	DIS	0.0	0.0	-7.5	-3.06	gold	tied	blue	blue	
V	125	0	2	tied	DIS	DIS	DIS	0.0	-11.11	-8.82	-3.37	tied	gold	gold	gold	
V	125	0	3	tied	DIS	TGS	TGS	0.0	-11.11	3.03	1.03	tied	tied	blue	blue	
V	125	0	4	tied	tied	DIS	DIS	0.0	0.0	-8.33	-5.15	tied	tied	blue	blue	
V	125	25	1	tied	tied	DIS	DIS	0.0	0.0	-15.22	-6.73	gold	tied	gold	gold	
V	125	25	2	TGS	DIS	DIS	DIS	12.5	-11.11	-9.76	-4.17	gold	gold	gold	gold	
V	125	25	3	tied	DIS	TGS	DIS	0.0	-11.11	4.65	-1.94	gold	blue	blue	blue	
V	125	25	4	tied	tied	DIS	DIS	0.0	0.0	-8.7	-5.66	gold	tied	gold	gold	
V	125	50	1	tied	DIS	DIS	DIS	0.0	-11.11	-11.67	-6.84	gold	tied	gold	blue	
V	125	50	2	tied	DIS	DIS	DIS	0.0	-11.11	-12.24	-5.77	gold	gold	gold	gold	
V	125	50	3	tied	DIS	DIS	DIS	0.0	-11.11	-4.0	-1.75	gold	tied	gold	tied	
V	125	50	4	tied	tied	DIS	DIS	0.0	0.0	-9.88	-7.69	gold	tied	gold	blue	

Benchmark Problem				DISASTER or TGS				Percent Δ				Blue or Gold				Dual
Identification				MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	
A	105	0	1	TGS	tied	TGS	TGS	25.0	0.0	10.34	3.66	gold	tied	gold	gold	
A	105	0	2	tied	tied	TGS	TGS	0.0	0.0	6.67	1.22	tied	tied	blue	blue	
A	105	0	3	tied	DIS	tied	tied	0.0	-11.11	0.0	0.0	blue	blue	blue	blue	
A	105	0	4	TGS	DIS	tied	tied	25.0	-11.11	0.0	0.0	tied	gold	blue	blue	
A	105	25	1	tied	tied	DIS	DIS	0.0	0.0	-5.26	-2.2	gold	tied	gold	gold	NO
A	105	25	2	tied	tied	DIS	DIS	0.0	0.0	-11.54	-3.23	tied	tied	gold	gold	NO
A	105	25	3	tied	tied	DIS	DIS	0.0	0.0	-10.71	-3.37	tied	tied	gold	gold	NO
A	105	25	4	tied	tied	DIS	DIS	0.0	0.0	-2.78	-1.11	gold	tied	gold	gold	NO
A	105	50	1	tied	tied	DIS	DIS	0.0	0.0	-1.96	-0.96	gold	tied	gold	gold	NO
A	105	50	2	tied	tied	DIS	DIS	0.0	0.0	-5.26	-1.9	gold	tied	gold	gold	NO
A	105	50	3	tied	tied	DIS	DIS	0.0	0.0	-2.27	-0.95	gold	tied	gold	gold	NO
A	105	50	4	tied	tied	DIS	DIS	0.0	0.0	-1.96	-0.95	gold	tied	gold	gold	NO
A	115	0	1	tied	tied	DIS	DIS	0.0	0.0	-3.13	-1.18	gold	tied	gold	gold	
A	115	0	2	DIS	DIS	tied	tied	-33.33	-12.5	0.0	0.0	blue	gold	blue	blue	
A	115	0	3	tied	tied	DIS	DIS	0.0	0.0	-9.09	-2.41	tied	tied	gold	gold	
A	115	0	4	tied	DIS	DIS	DIS	0.0	-11.11	-22.22	-7.41	tied	tied	blue	blue	
A	115	25	1	tied	tied	DIS	DIS	0.0	0.0	-4.55	-2.06	gold	tied	gold	gold	NO
A	115	25	2	tied	tied	DIS	DIS	0.0	0.0	-9.38	-3.03	tied	tied	gold	gold	NO
A	115	25	3	tied	tied	tied	tied	0.0	0.0	0.0	0.0	gold	tied	gold	gold	NO
A	115	25	4	tied	tied	DIS	DIS	0.0	0.0	-4.76	-2.08	gold	tied	gold	gold	NO
A	115	50	1	tied	tied	tied	tied	0.0	0.0	0.0	0.0	tied	tied	gold	gold	NO
A	115	50	2	tied	tied	DIS	DIS	0.0	0.0	-2.08	-0.87	tied	tied	gold	gold	NO
A	115	50	3	tied	tied	tied	tied	0.0	0.0	0.0	0.0	tied	tied	gold	gold	NO
A	115	50	4	tied	tied	DIS	DIS	0.0	0.0	-1.69	-0.88	gold	tied	gold	gold	NO
A	125	0	1	tied	tied	DIS	DIS	0.0	0.0	-8.11	-3.33	gold	tied	gold	gold	
A	125	0	2	tied	DIS	DIS	DIS	0.0	-11.11	-3.85	-1.08	blue	blue	gold	gold	
A	125	0	3	tied	tied	tied	tied	0.0	0.0	0.0	0.0	tied	tied	blue	blue	
A	125	0	4	tied	tied	DIS	DIS	0.0	0.0	-2.78	-1.11	tied	tied	blue	blue	
A	125	25	1	tied	tied	DIS	DIS	0.0	0.0	-1.96	-0.96	gold	tied	gold	gold	NO
A	125	25	2	tied	tied	DIS	DIS	0.0	0.0	-5.26	-1.9	gold	tied	gold	gold	NO
A	125	25	3	tied	tied	DIS	DIS	0.0	0.0	-2.27	-0.95	gold	tied	gold	gold	NO
A	125	25	4	tied	tied	DIS	DIS	0.0	0.0	-1.96	-0.95	gold	tied	gold	gold	NO
A	125	50	1	tied	tied	DIS	DIS	0.0	0.0	-1.45	-0.82	gold	tied	gold	gold	NO
A	125	50	2	tied	tied	DIS	DIS	0.0	0.0	-1.75	-0.81	gold	tied	gold	gold	NO
A	125	50	3	tied	tied	tied	tied	0.0	0.0	0.0	0.0	gold	tied	gold	gold	NO
A	125	50	4	tied	tied	DIS	DIS	0.0	0.0	-2.94	-1.64	gold	tied	gold	gold	NO

Benchmark Problem				DISASTER or TGS				%Δ				Blue or Gold				Dual
Identification				MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	MTD	%TJ	TDL	AFT	
T	105	0	1	tied	tied	tied	tied	0.0	0.0	0.0	0.0	blue	blue	blue	blue	
T	105	0	2	TGS	DIS	tied	tied	16.67	-12.5	0.0	0.0	gold	blue	blue	tied	
T	105	0	3	tied	tied	tied	DIS	0.0	0.0	0.0	-4.35	tied	gold	gold	gold	
T	105	0	4	TGS	DIS	DIS	DIS	25.0	-25.0	-9.09	-2.94	gold	gold	gold	gold	
T	105	25	1	tied	DIS	DIS	DIS	0.0	-25.0	-12.9	-5.88	gold	blue	gold	gold	
T	105	25	2	tied	DIS	DIS	DIS	0.0	-25.0	-10.87	-5.62	gold	blue	blue	blue	
T	105	25	3	tied	tied	DIS	DIS	0.0	0.0	-5.26	-5.06	gold	tied	gold	gold	NO
T	105	25	4	TGS	DIS	DIS	DIS	16.67	-14.29	-4.35	-2.5	gold	gold	gold	gold	
T	105	50	1	tied	DIS	DIS	DIS	0.0	-11.11	-11.11	-6.06	gold	gold	gold	gold	
T	105	50	2	tied	DIS	DIS	DIS	0.0	-25.0	-6.35	-4.76	gold	gold	tied	gold	NO
T	105	50	3	tied	tied	DIS	DIS	0.0	0.0	-3.23	-4.4	gold	tied	gold	gold	NO
T	105	50	4	DIS	DIS	DIS	DIS	-14.29	-12.5	-18.18	-7.78	gold	gold	gold	gold	NO
T	115	0	1	tied	DIS	DIS	DIS	0.0	-14.29	-3.57	-2.44	blue	blue	tied	gold	
T	115	0	2	TGS	DIS	DIS	DIS	14.29	-12.5	-2.44	-1.19	gold	blue	blue	blue	
T	115	0	3	tied	tied	tied	DIS	0.0	0.0	0.0	-4.05	tied	tied	gold	gold	
T	115	0	4	TGS	DIS	DIS	DIS	20.0	-16.67	-6.25	-2.74	gold	gold	gold	gold	
T	115	25	1	tied	DIS	DIS	DIS	0.0	-25.0	-15.38	-7.53	gold	blue	gold	gold	
T	115	25	2	tied	DIS	DIS	DIS	0.0	-25.0	-8.77	-5.0	gold	blue	blue	tied	NO
T	115	25	3	tied	DIS	DIS	DIS	0.0	-16.67	-7.69	-5.81	gold	blue	gold	gold	NO
T	115	25	4	DIS	DIS	DIS	DIS	-16.67	-14.29	-21.43	-8.24	gold	gold	gold	gold	NO
T	115	50	1	tied	DIS	DIS	DIS	0.0	-11.11	-12.73	-7.34	gold	gold	gold	gold	
T	115	50	2	tied	DIS	DIS	DIS	0.0	-11.11	-6.85	-4.31	gold	tied	blue	blue	NO
T	115	50	3	tied	DIS	tied	DIS	0.0	-12.5	0.0	-2.91	gold	blue	gold	gold	NO
T	115	50	4	tied	DIS	DIS	DIS	0.0	-11.11	-9.09	-4.95	gold	gold	gold	gold	NO
T	125	0	1	tied	DIS	DIS	DIS	0.0	-28.57	-9.38	-3.45	blue	blue	blue	tied	
T	125	0	2	TGS	DIS	DIS	DIS	12.5	-12.5	-2.22	-1.14	gold	blue	blue	blue	
T	125	0	3	tied	DIS	tied	DIS	0.0	-20.0	0.0	-3.8	tied	blue	gold	gold	
T	125	0	4	TGS	DIS	DIS	DIS	16.67	-16.67	-4.76	-2.56	gold	gold	gold	gold	
T	125	25	1	tied	DIS	DIS	DIS	0.0	-11.11	-11.11	-6.06	gold	gold	gold	gold	
T	125	25	2	TGS	DIS	DIS	DIS	9.09	-25.0	-9.84	-5.77	gold	blue	blue	blue	
T	125	25	3	tied	DIS	DIS	DIS	0.0	-16.67	-3.23	-4.4	gold	blue	gold	gold	NO
T	125	25	4	DIS	DIS	DIS	DIS	-14.29	-12.5	-18.18	-7.78	gold	gold	gold	gold	NO
T	125	50	1	tied	DIS	DIS	DIS	0.0	-11.11	-8.06	-5.17	gold	gold	gold	gold	
T	125	50	2	tied	DIS	DIS	DIS	0.0	-11.11	-8.97	-6.67	gold	tied	gold	gold	NO
T	125	50	3	tied	DIS	DIS	DIS	0.0	-12.5	-2.08	-3.7	gold	blue	gold	gold	NO
T	125	50	4	tied	DIS	DIS	DIS	0.0	-11.11	-10.0	-5.61	gold	gold	gold	gold	NO

Bibliography

Carlson, Barak J. and Christopher A. Lettiere. *A Comparison of the DISASTER™ Scheduling Software With a Simultaneous Scheduling Algorithm for Minimizing Maximum Tardiness in Job Shops.* MS thesis, AFIT/GSM/LAS/93S-3. Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1993 (AD-A276192).

Conway, Richard W., and others. *Theory of Scheduling.* Reading MA. Addison-Wesley Publishing Company, 1967.

Gargeya, Vidyaranya B. "An Evaluation of Resource Constraint Measures in a Dual Constrained Job Environment," *Proceedings of the 1992 Annual Meeting of the Decision Sciences Institute.* 1538-1540.

Goal System Version 2.2, SCHEDULE Users Guide, TOC Center Inc. (1995).

Goldratt, Eliyahu M. and Jeff Cox. *The Goal: A Process of Ongoing Improvement.* (Revised Edition). Croton-on-Hudson NY: North River Press, 1986.

Goldratt, Eliyahu M. and Robert E. Fox. *The Race.* Croton-on-Hudson NY: North River Press, 1986.

Goldratt, Eliyahu M. *The Haystack Syndrome: Sifting Information Out of the Data Ocean.* Croton-on-Hudson NY: North River Press, 1990.

Graves, Stephen. "A Review of Production Scheduling," *Operations Research, 29:* 646-675 (July/August 1981).

Guide, V. Daniel R., Jr., and Gerald A. Ghiselli. "Implementation of Drum-Buffer-Rope at a Military Rework Depot Engine Works," *Production and Inventory Management Journal, 36:* 79-83 (Third Quarter 1995).

James, Stewart W. and Bruno A. Mediate Jr. *Benchmark Production Scheduling Problems for Job Shops With Interactive Constraints.* MS thesis, AFIT/GSM/LAS/93S-3. Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1993 (AD-A273962).

Lalsare, Parag, and Swapan Sen. "Evaluating Backward Scheduling and Sequencing Rules for an Assembly Shop Environment," *Production and Inventory Management Journal, 36:* 71-78 (Fourth Quarter 1995).

Roy, R., and S. E. Meikle. "The Role of Discrete Event Simulation Techniques in Finite Capacity Scheduling," *Journal of the Operational Research Society, 46:* 1310-1321 (November 1995).

Schragenheim, Eli, and Boaz Ronen. "Drum-Buffer-Rope Shop Floor Control," *Production and Inventory Management Journal, 31:* 18-23 (Third Quarter 1990).

Simons, Jacob V. Jr. and others. *Formulation and Solution of the Drum-Buffer-Rope Constraint Scheduling Problem (DBRCSP).* Final report, AFIT-LA-TM-95-2. Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1995.

Simons, Jacob V. Jr. and Wendell P. Simpson, III. *An Explication and Characterization of the Goal System (formerly DISASTER™) Approach to Multiple Constraint Scheduling for Drum-Buffer-Rope (DBR) Production Management Systems.* Final report, AFIT-LA-TM-95-1. Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1995.

-----. "An Exposition of Multiple Constraint Scheduling as Implemented in THE GOAL SYSTEM (formerly DISASTER™)," *Production and Operations Management,* (forthcoming, 1996).

Spencer, Michael S. and James F. Cox III. "Master Production Scheduling Development in a Theory of Constraints Environment," *Production and Inventory Management Journal, 36:* 8-14 (First Quarter 1995).

-----. "Optimum Production Technology (OPT) and the Theory of Constraints (TOC): Analysis and Genealogy," *International Journal of Production Research, 33:* 1495-1504 (June 1995).

Umble, Michael M. and Mokshagundam L. Srikanth. *Synchronous Manufacturing: Principles for World Class Excellence.* Cincinnati: South-Western Publishing, 1990.

Vita

Captain Michael D. Stephens was born on 21 September 1962 in Harrisburg, Pennsylvania. He graduated from Central Dauphin High School in 1980, and enlisted in the United States Air Force. His first assignment was to George AFB, California, as an Electronic Warfare Systems Specialist. Over the next twelve years he was assigned to Shaw AFB, South Carolina; Offutt AFB, Nebraska; and twice to Osan AB, Korea. In October 1988, he earned an Associate of Applied Science degree from the Community College of the Air Force (majoring in Avionics Systems Technology), and in August 1990, he earned a Bachelor of Science degree from the Southern Illinois University at Carbondale (majoring in Electronics Management). He was commissioned through the Officers Training Group in July 1992. His next assignment was to Tyndall AFB, Florida, as an F-15 Aircraft Maintenance and Munitions Officer. There he served as Flight Commander of 325th Maintenance Squadron Propulsion Flight, and Flight Commander of the 2d Fighter Squadron Sortie Support and Sortie Generation Flights. He entered the Graduate School of Logistics and Acquisition Management, Air Force Institute of Technology, in May of 1995. Following graduation he will be assigned to the Human Systems Center, Brooks AFB, Texas, as an Acquisition Logistics Manager.

Permanent Address: 4912 Follins Court
Harrisburg PA 17112

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1996	Master's Thesis

4. TITLE AND SUBTITLE
EVALUATION OF *THE GOAL SYSTEM*™ VERSION 2.2 SOLUTION METHOD FOR INTERACTIVE CONSTRAINT SCHEDULING SITUATIONS

5. FUNDING NUMBERS

6. AUTHOR(S)
Michael D. Stephens, Captain, USAF

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)
Air Force Institute of Technology
2750 P Street
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GLM/LAL/96S-7

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
THE GOAL SYSTEM™ version 2.2 is the latest in a lineage that includes Optimized Production Technology (OPT) and *DISASTER*™. Earlier work with *DISASTER*™ revealed potential shortcomings with sequential schedule-building algorithms when multiple interactive constraints exist. Since *THE GOAL SYSTEM*™ version 2.2 has a capacity for simultaneous schedule-building, this study evaluated differences between the two algorithms. Using benchmark scheduling problems developed during the earlier evaluation of *DISASTER*™, a set of *THE GOAL SYSTEM*™ solutions was created and compared quantitatively to both *DISASTER*™ solutions and solutions which optimally minimize maximum tardiness. A broad set of performance measurement criteria were also used to obtain a more comprehensive evaluation of the solutions. Performance of *THE GOAL SYSTEM*™ was quite good with respect to maximum tardiness. Performance with respect to average flow-time, percentage of tardy jobs, and total days late for a set of job orders was markedly poorer than the *DISASTER*™ solutions. The results were unexpected, since the simultaneous scheduling algorithm is less restricted in its options for schedule creation. The author concluded that the simultaneous feature of *THE GOAL SYSTEM*™ was better suited for conflict resolution during an iterative process than as a stand-alone scheduling algorithm.

14. SUBJECT TERMS
Theory of Constraints, production scheduling, Drum-Buffer-Rope, finite capacity scheduling, scheduling software

15. NUMBER OF PAGES
75

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

AFIT Control Number AFIT/GLM/LAL/96S-7

AFIT RESEARCH ASSESSMENT

The purpose of this questionnaire is to determine the potential for current and future applications of AFIT thesis research. **Please return completed questionnaire** to: AIR FORCE INSTITUTE OF TECHNOLOGY/LAC, 2950 P STREET, WRIGHT-PATTERSON AFB OH 45433-7765. Your response is **important.** Thank you.

1. Did this research contribute to a current research project? a. Yes b. No

2. Do you believe this research topic is significant enough that it would have been researched (or contracted) by your organization or another agency if AFIT had not researched it?
a. Yes b. No

3. **Please estimate** what this research would have cost in terms of manpower and dollars if it had been accomplished under contract or if it had been done in-house.

Man Years____________ $____________

4. Whether or not you were able to establish an equivalent value for this research (in Question 3), what is your estimate of its significance?

a. Highly Significant b. Significant c. Slightly Significant d. Of No Significance

5. Comments (Please feel free to use a separate sheet for more detailed answers and include it with this form):

______________________________ ______________________________
Name and Grade Organization

______________________________ ______________________________
Position or Title Address